MOVING AND COMPUTING

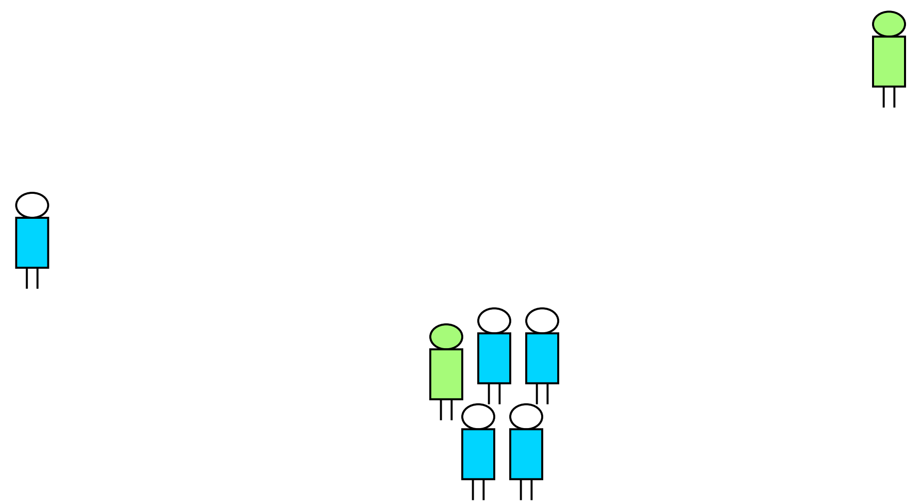
Nicola Santoro
Carleton University

Distributed Computations by Mobile Entities

Nicola Santoro
Carleton University

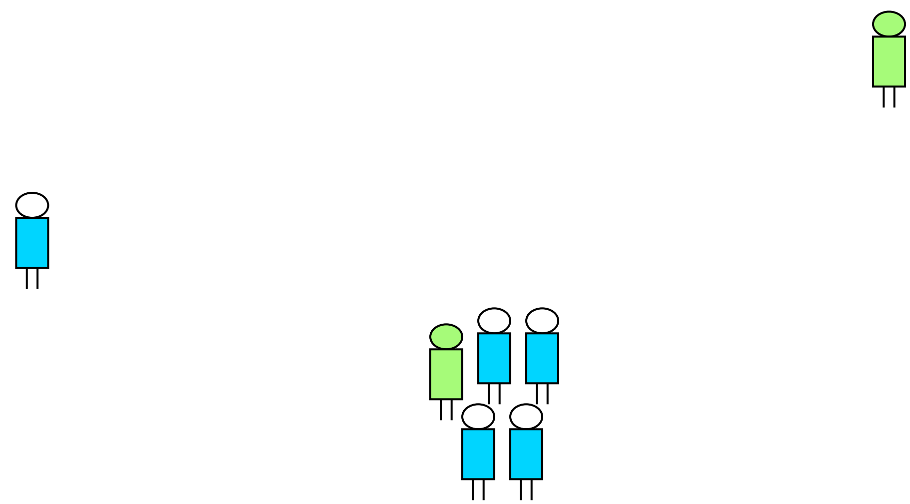
GRASTA/MAC Tutorial 2015

COMPUTATIONAL ENTITIES

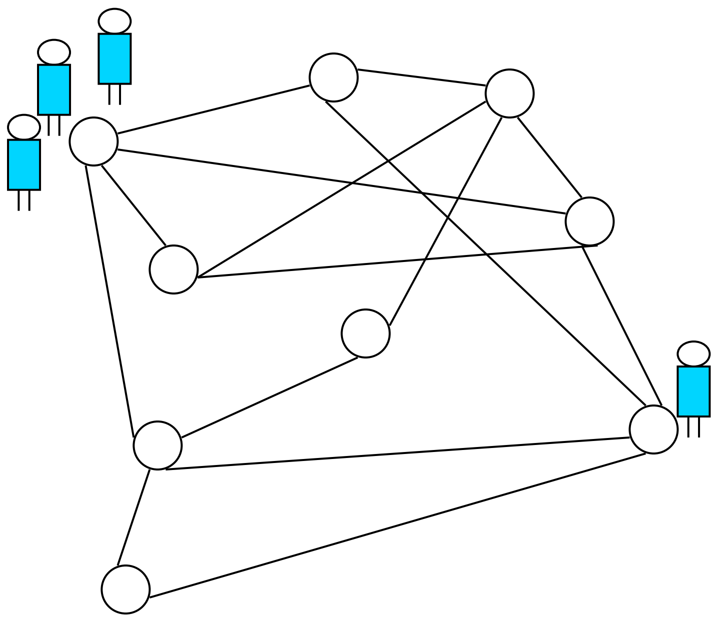


COMPUTATIONAL ENTITIES

OPERATE AND MOVE IN SOME SPACE



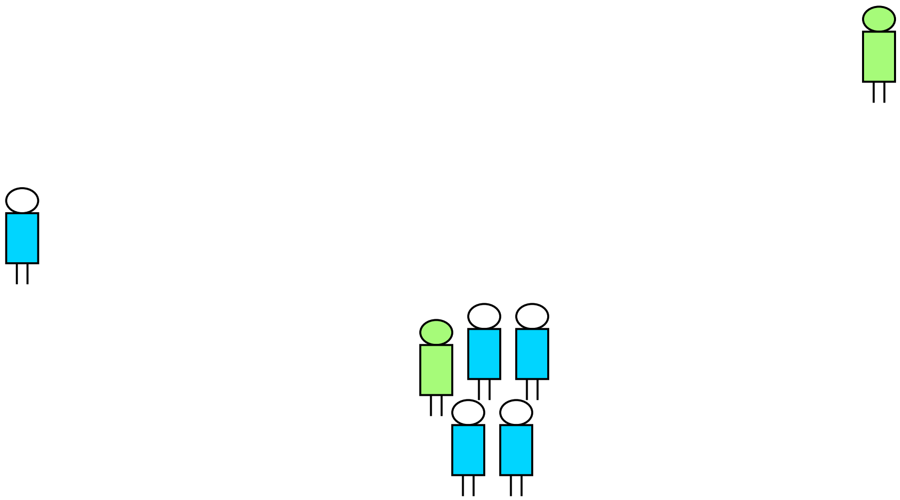
DISCRETE : NETSCAPE - GRAPH WORLD -



MOBILE AGENTS

CONTINUOUS : TERRAIN - 2D/3D SPACE -

MOBILE ROBOTS/SENSORS



MOVING AND COMPUTING

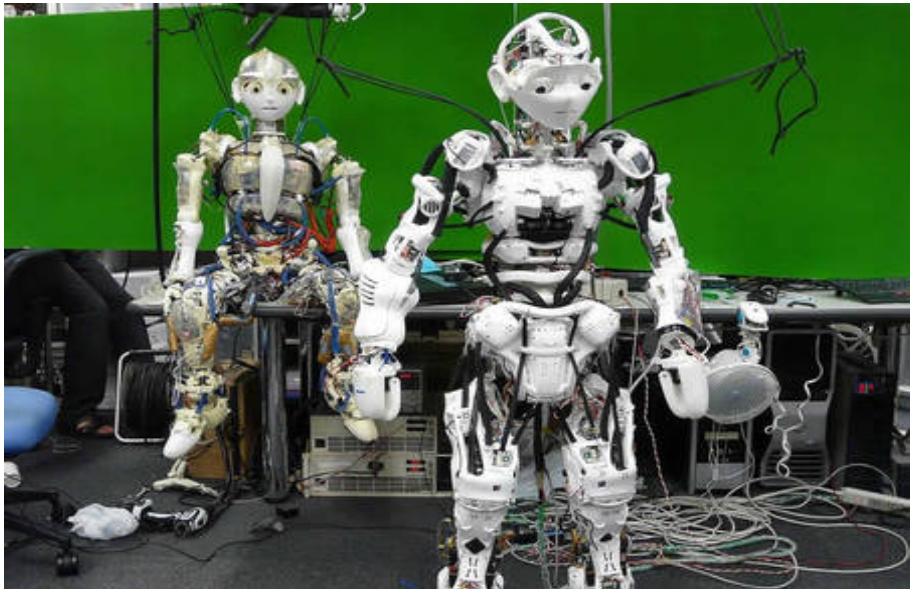
IN

CONTINUOUS SPACES

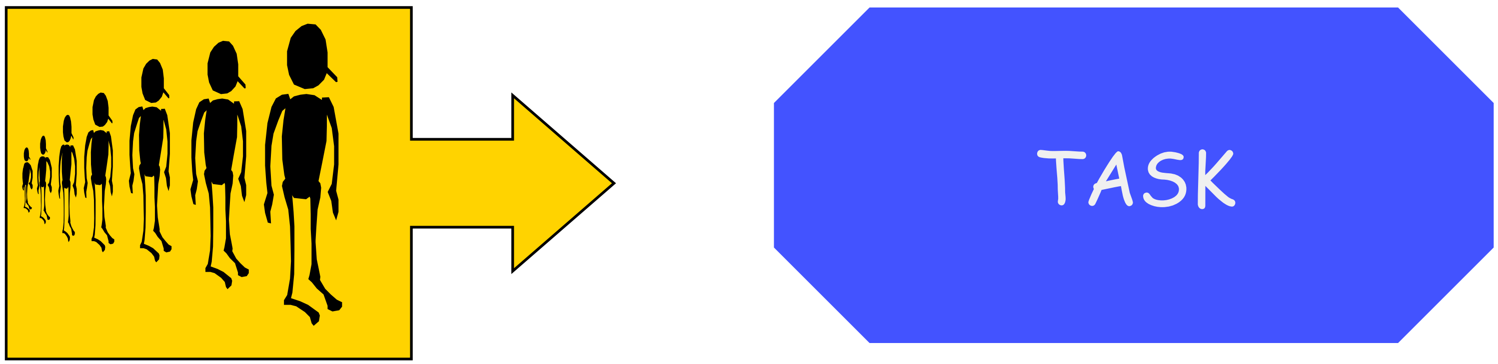
Some Motivations from Robotics

Some Motivations from Robotics

Few	⟷	Many
Complex		Very Simple
Specialized		Generic
Hetherogeneous		Identical
Coordinated		Autonomous



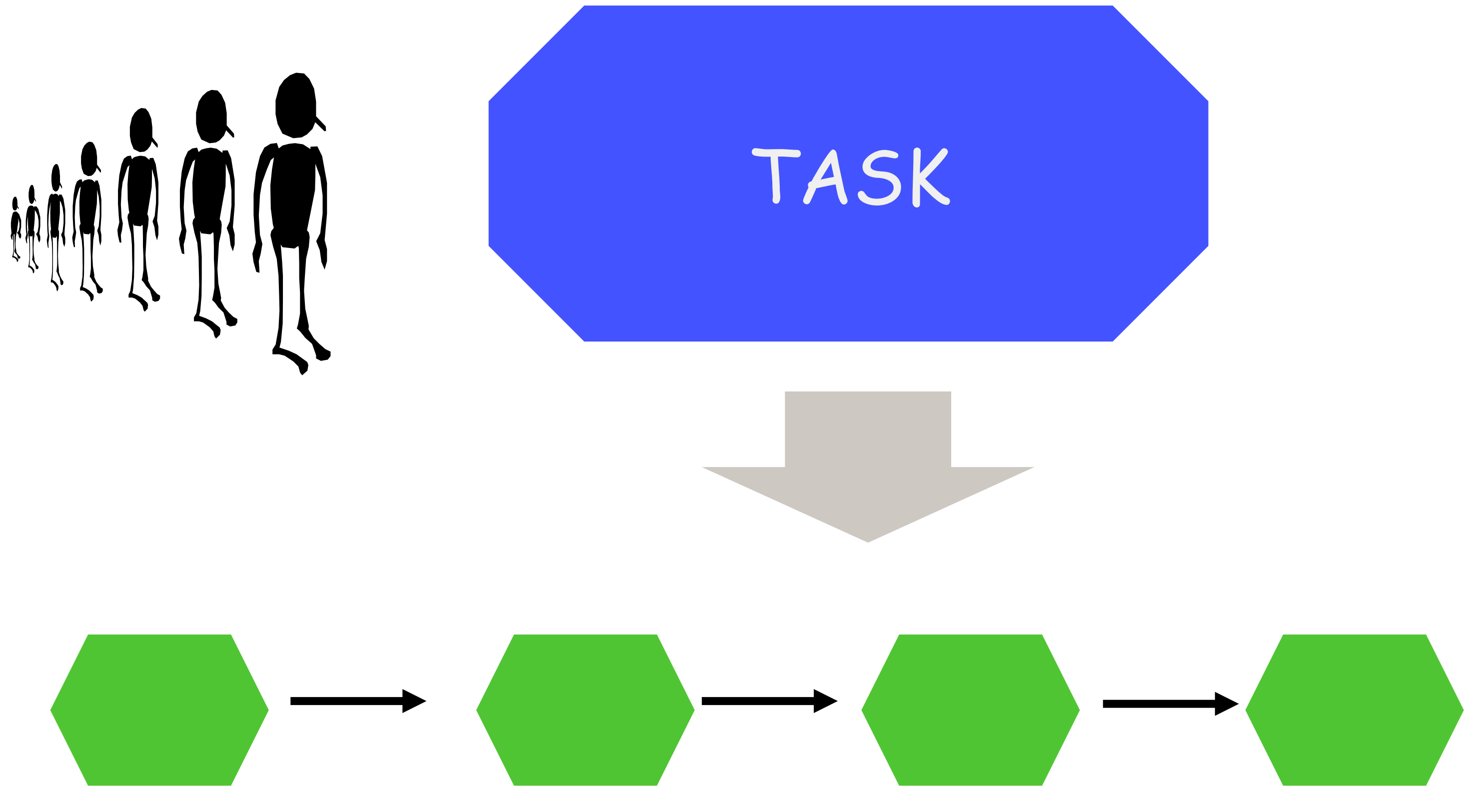
Some Motivations from Robotics



System (SWARM)

- A robot - **alone** - is computationally weak
- **Cooperation** of robots is essential to perform complex tasks

Some Motivations from Robotics



complex task → sequence of **basic tasks**

Basic Coordination Tasks

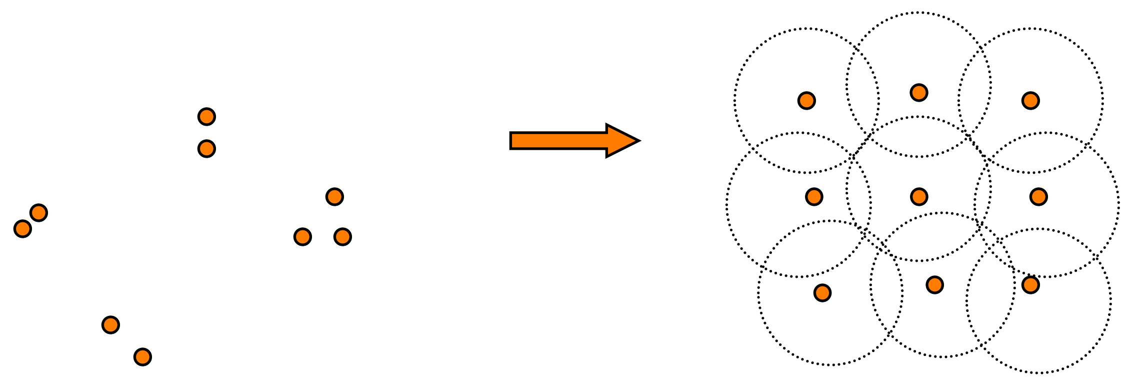
Gathering

Basic Coordination Tasks

Gathering

Alignment

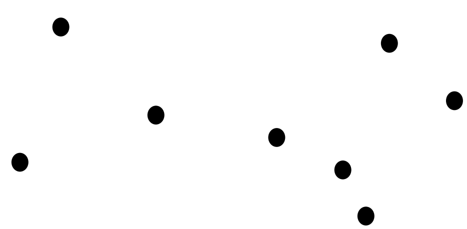
Scattering



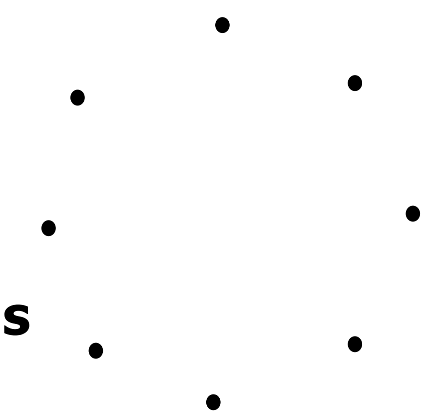
Basic Coordination Tasks

Pattern Formation

initial configuration → target pattern



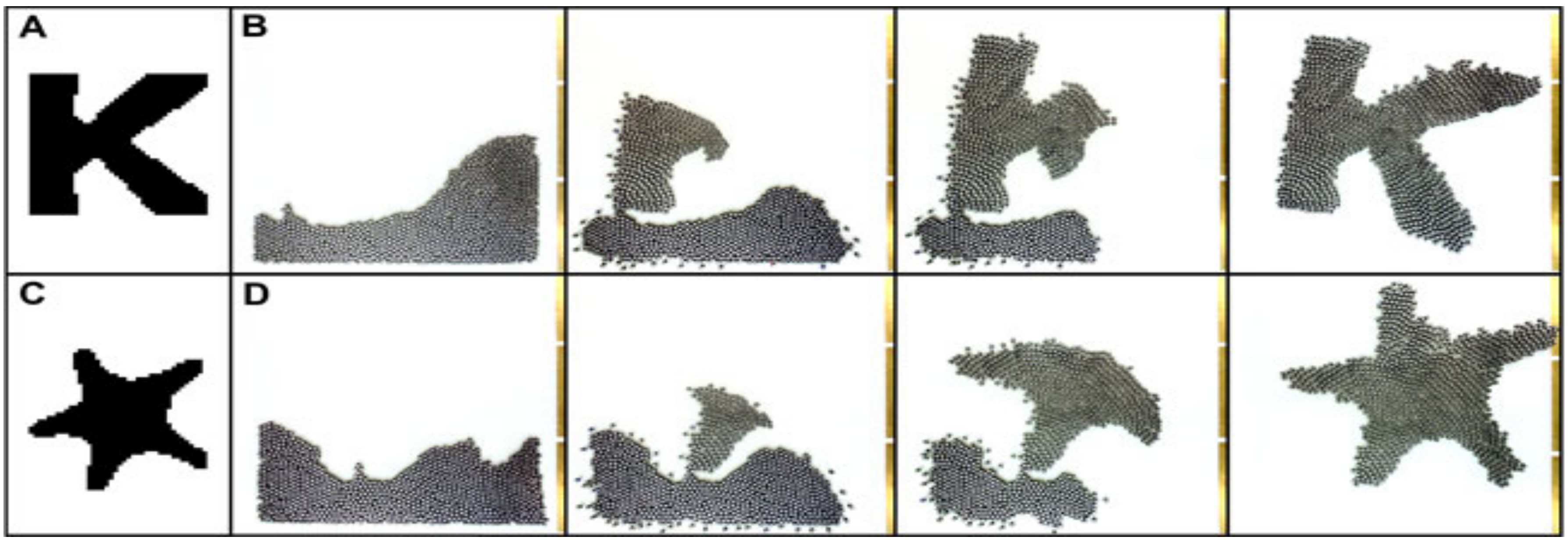
irrespective to
scaling,rotations,translations



Configuration C' **Pattern *P***

Basic Coordination Tasks

Pattern Formation



Main Research Question

How **weak** can each single robot be and still globally accomplish the given task ?

How much **local power** is necessary to perform **global computations** ?

COMPUTATIONAL MODEL

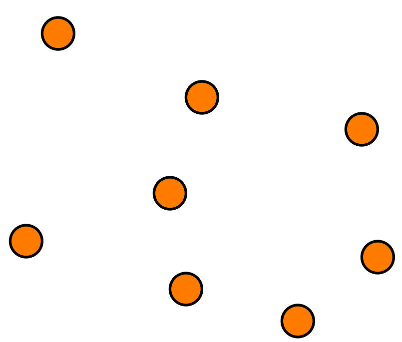
Mobile Robots

Autonomous (no central control)

Homogeneous (run same software)

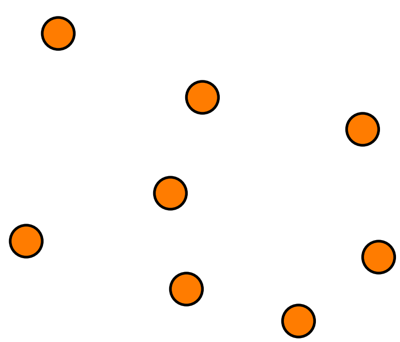
Identical (indistinguishable/anonymous)

Silent (no communication)



Mobile Robots

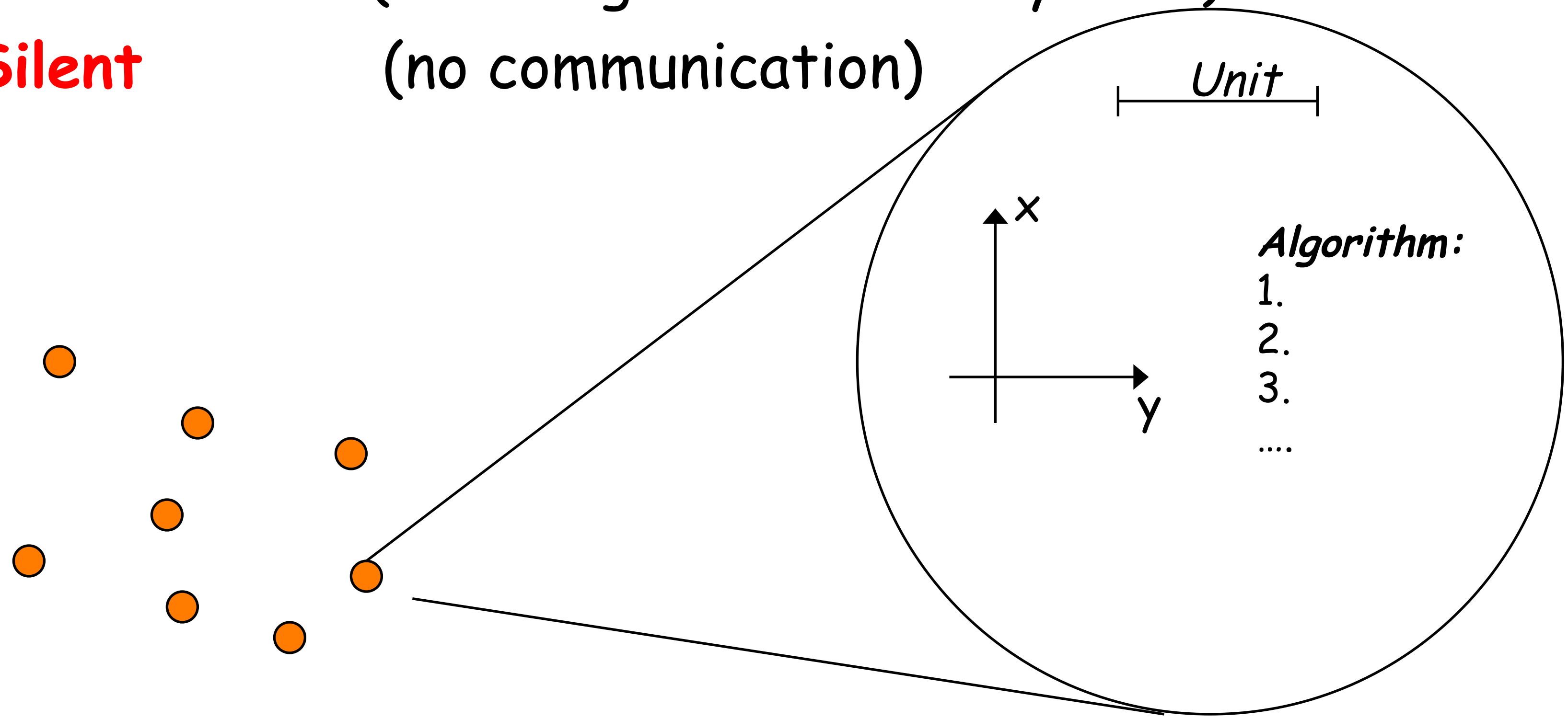
Autonomous (no central control)
Homogeneous (run same software)
Identical (indistinguishable/anonymous)
Silent (no communication)



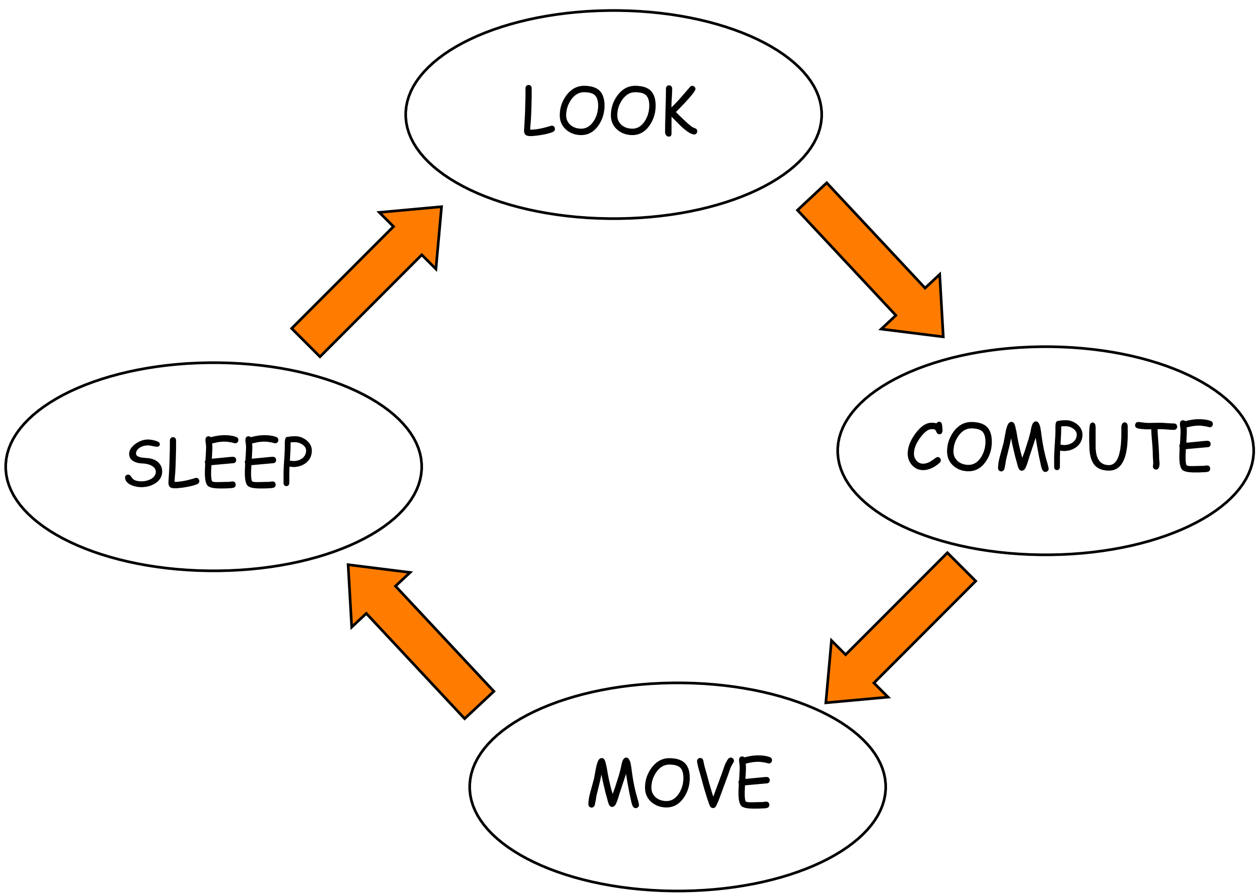
viewed as points.

Mobile Robots

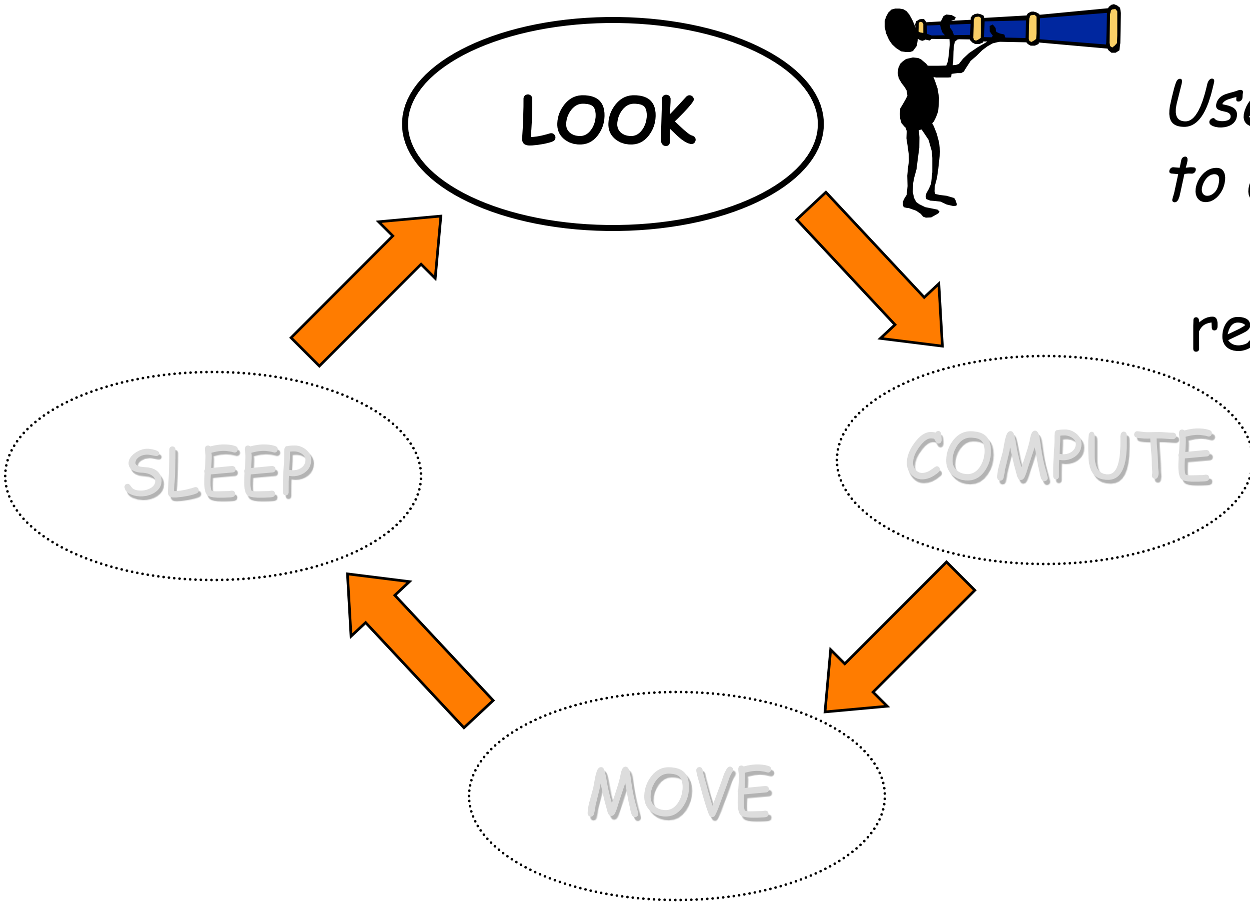
Autonomous (no central control)
Homogeneous (run same software)
Identical (indistinguishable/anonymous)
Silent (no communication)



Robot's behaviour : Life Cycle



Robot's behaviour : Life Cycle

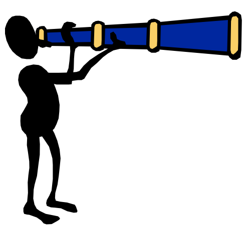


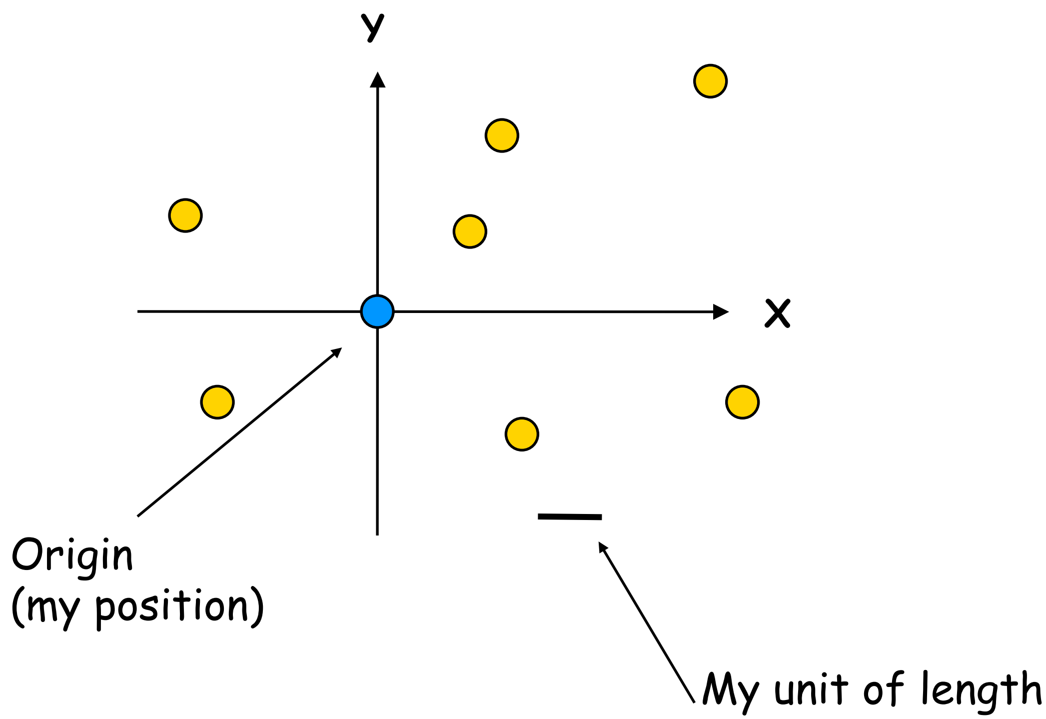
Uses its sensors to observe the world.

result = **SNAPSHOT** of the world

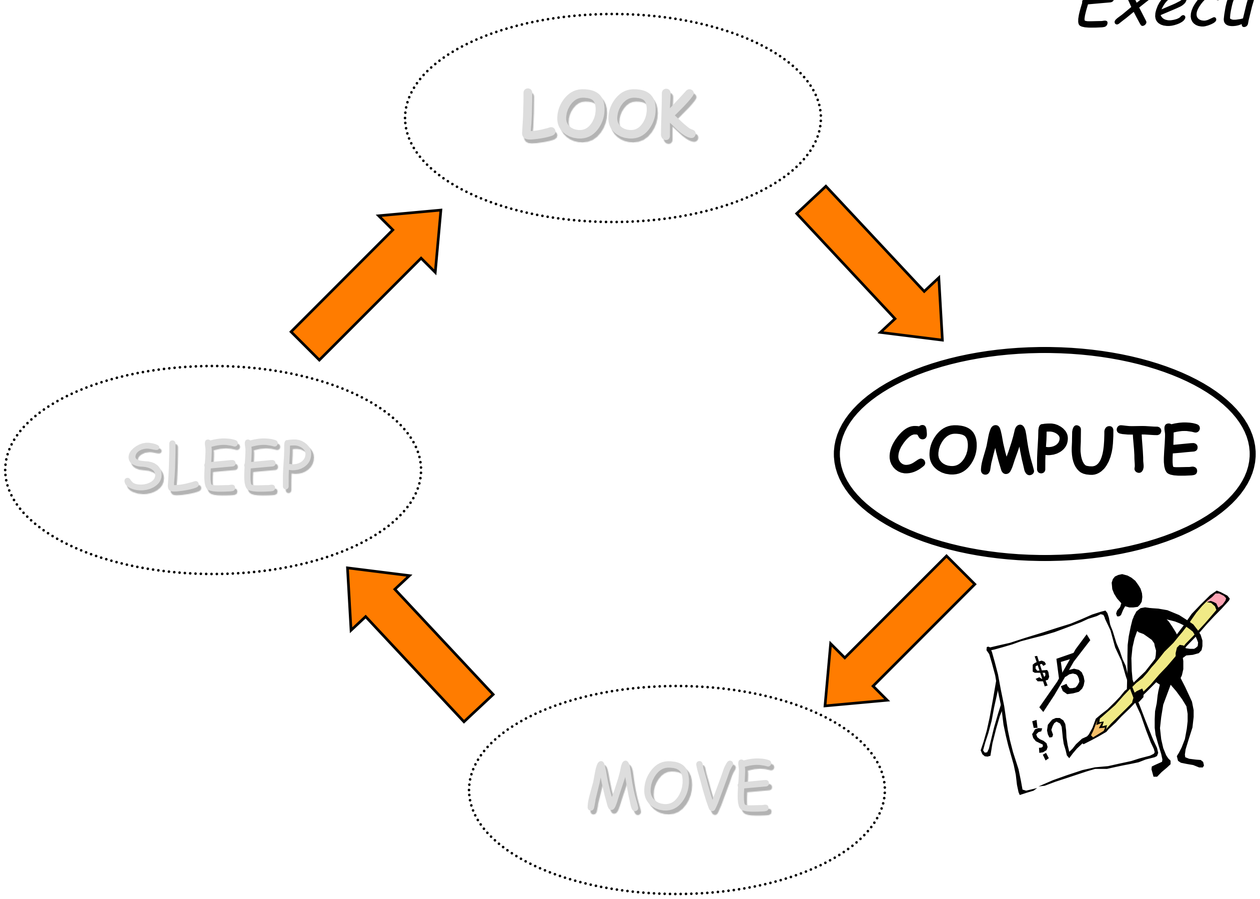
Snapshot

- returns the position of the other robots in terms of my local coordinate system





Robot's behaviour : Life Cycle

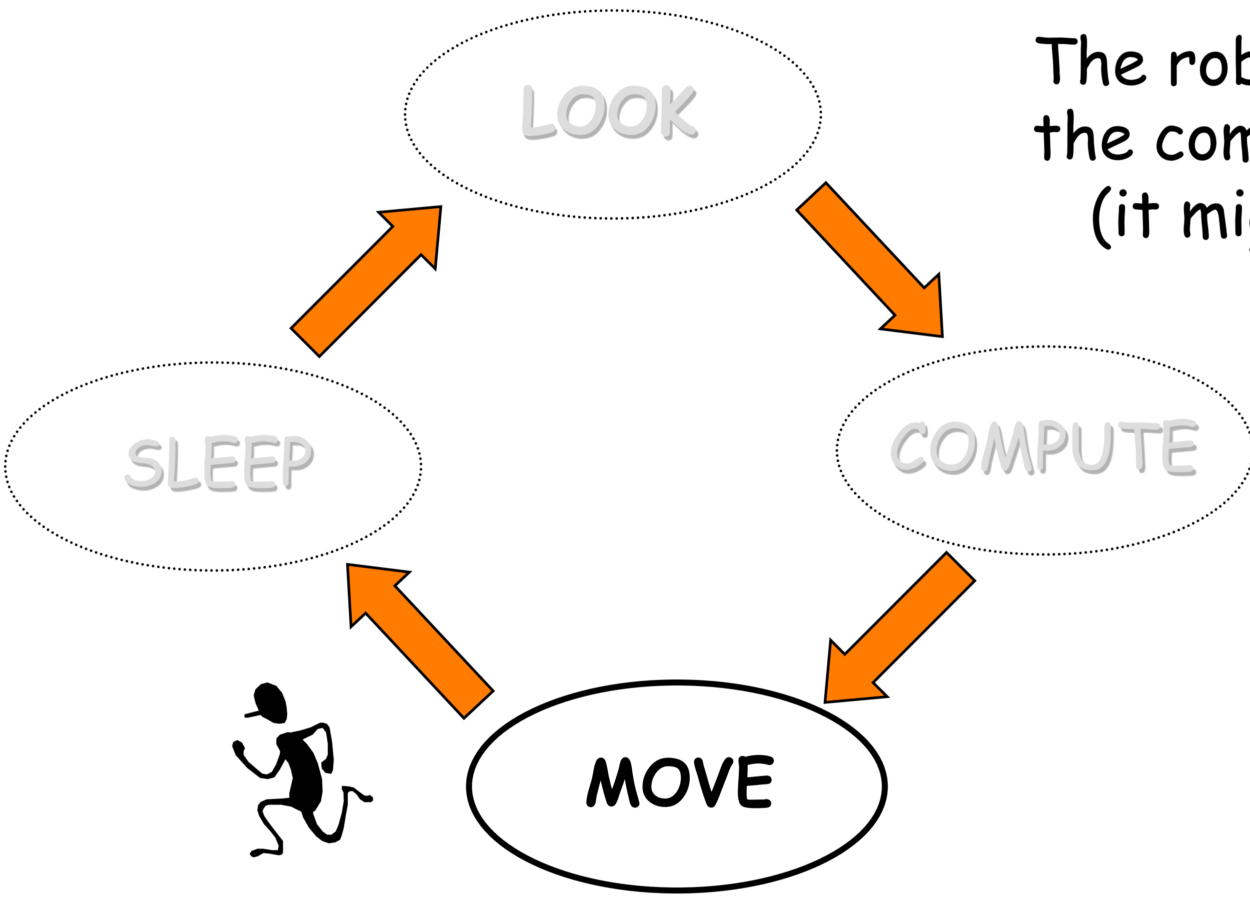


Executes the algorithm

input = snapshot

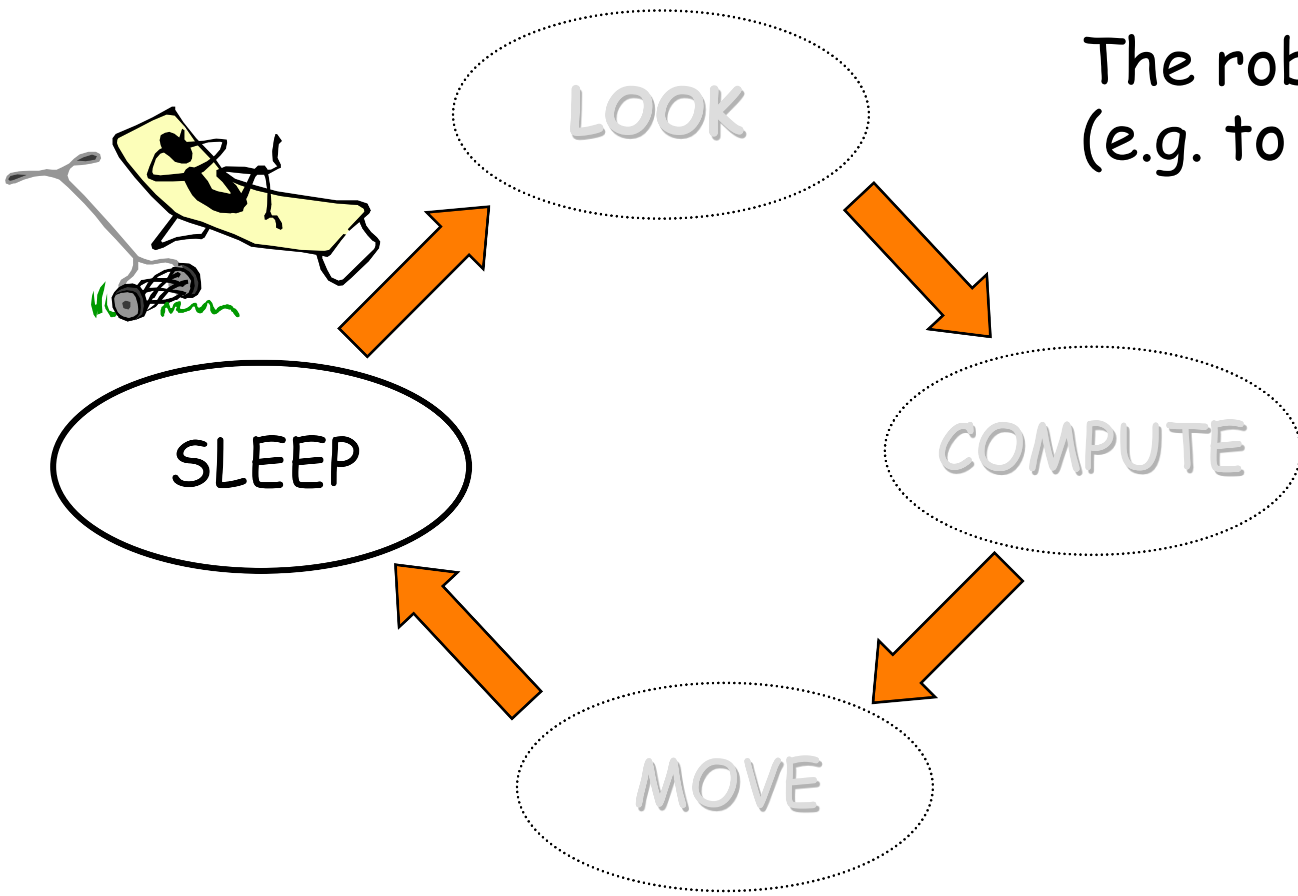
result = destination point

Robot's behaviour : Life Cycle



The robot moves *towards* the computed destination (it might not reach it)

Robot's behaviour : Life Cycle



The robot may be idle (e.g. to recharge battery)

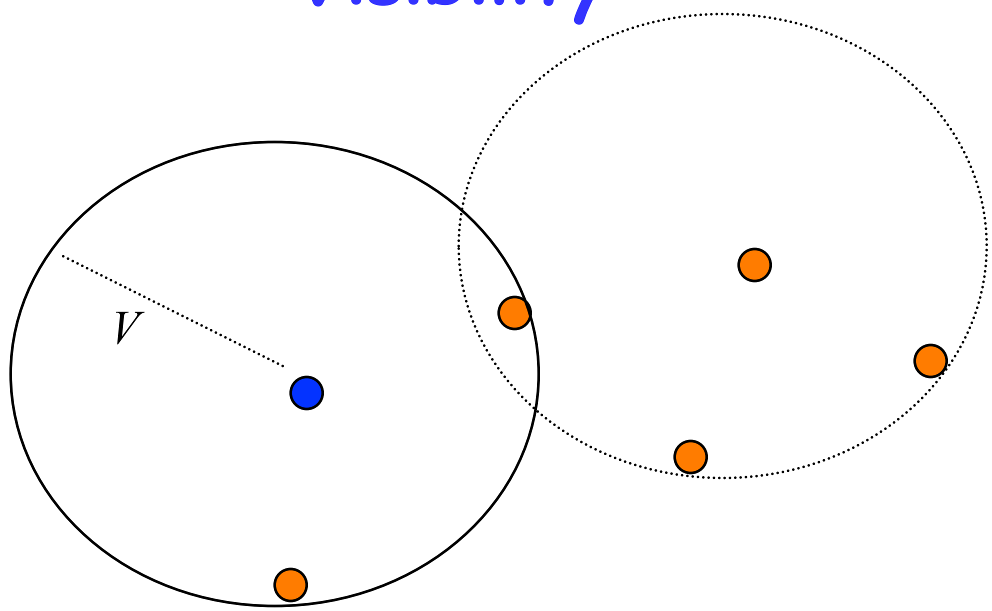
Crucial Factors

- Visibility

Crucial Factors

Limited

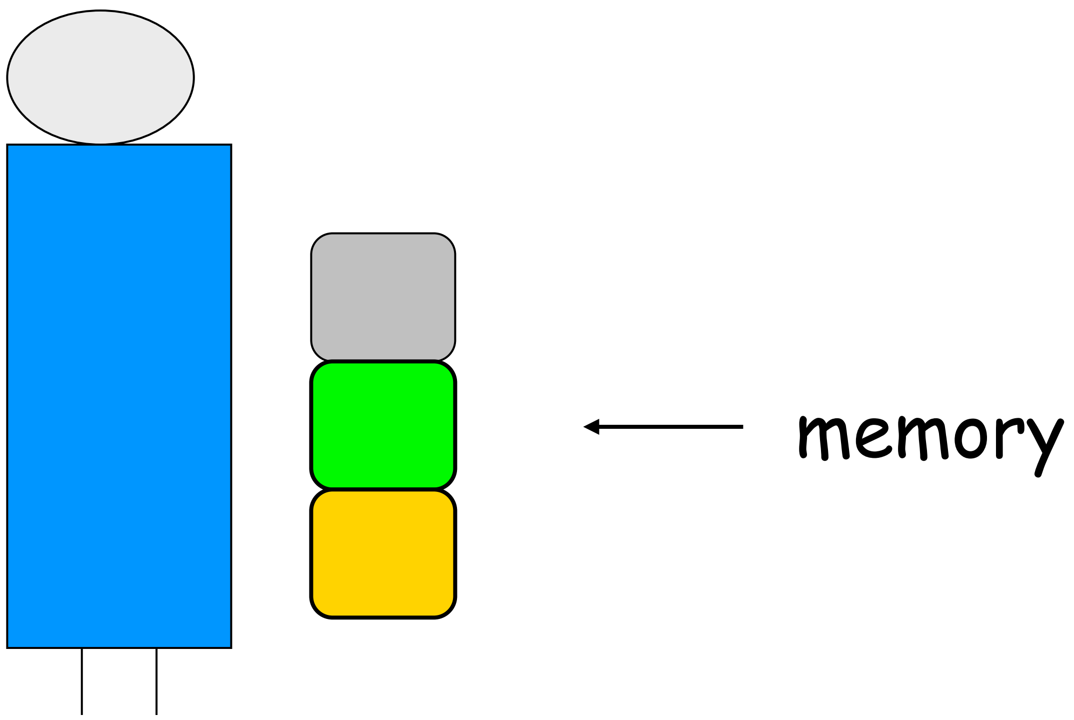
- **Visibility**



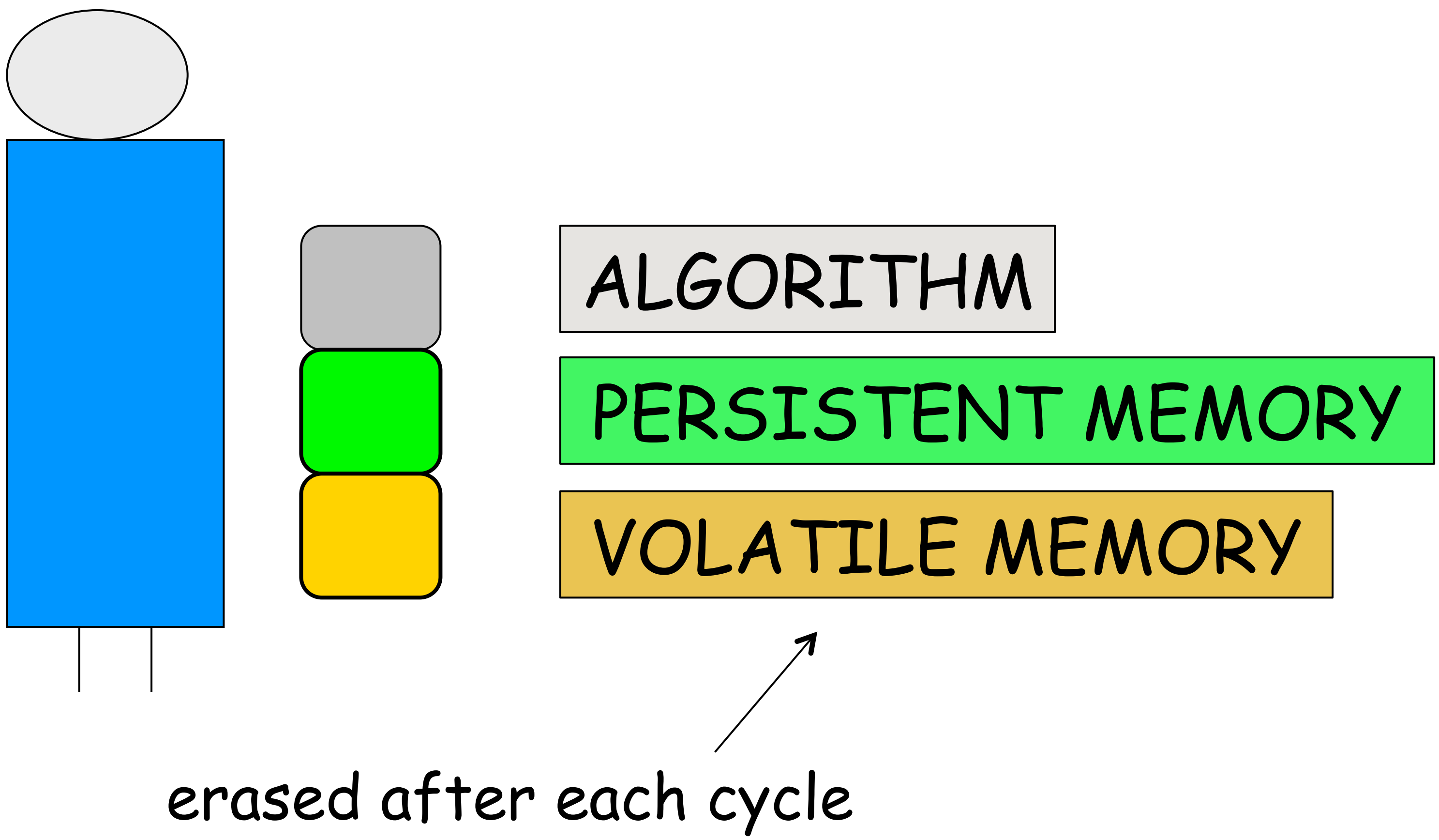
Crucial Factors

- **Visibility** Global → **Limited**
- Memory

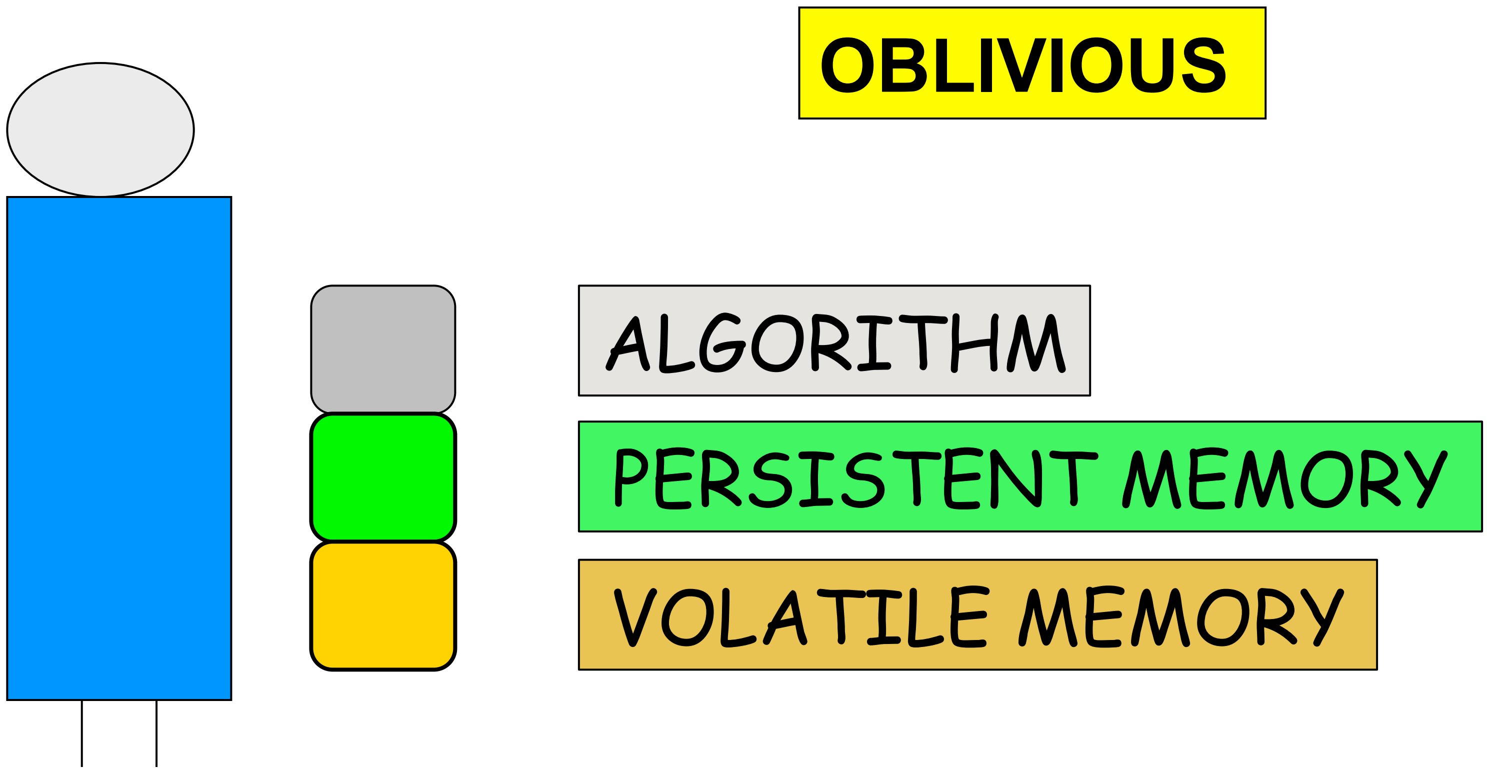
Memory



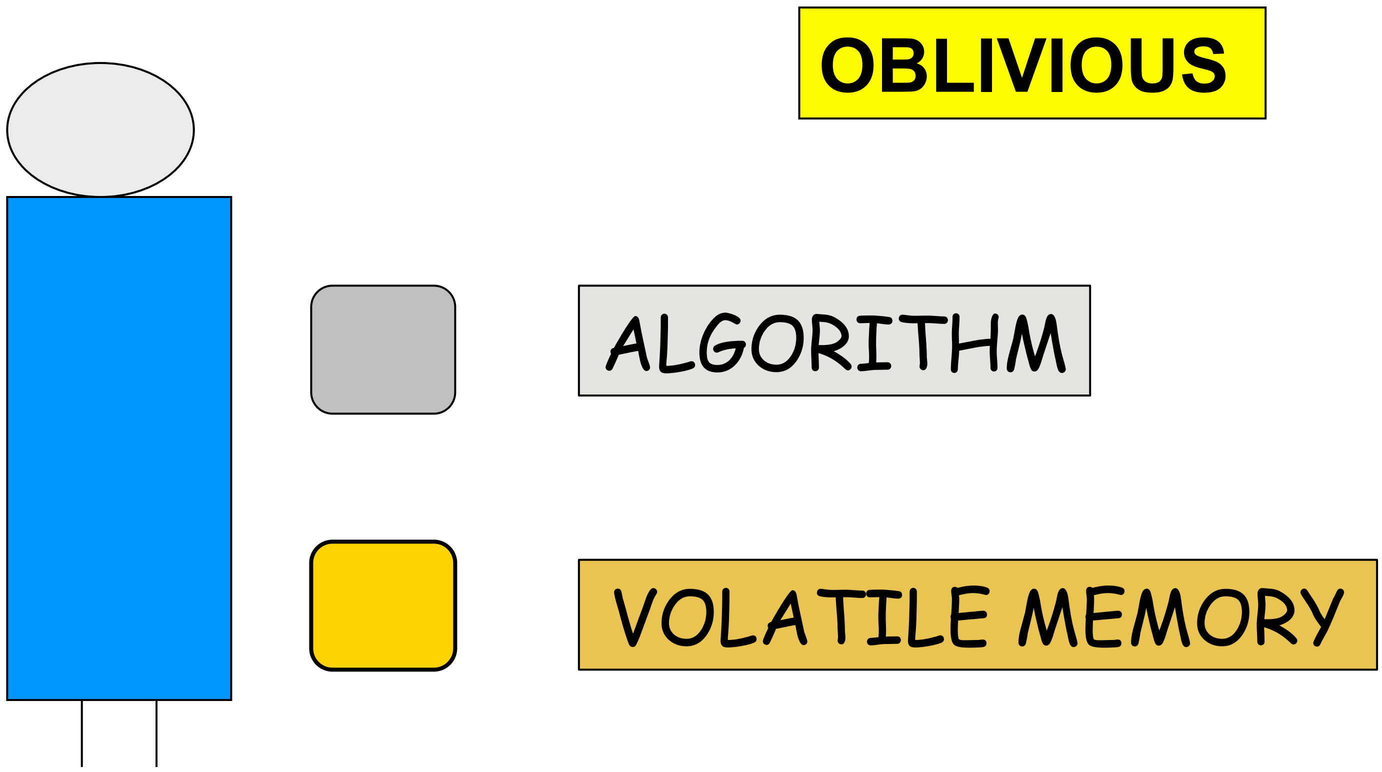
Memory


ALGORITHM
PERSISTENT MEMORY
VOLATILE MEMORY
erased after each cycle

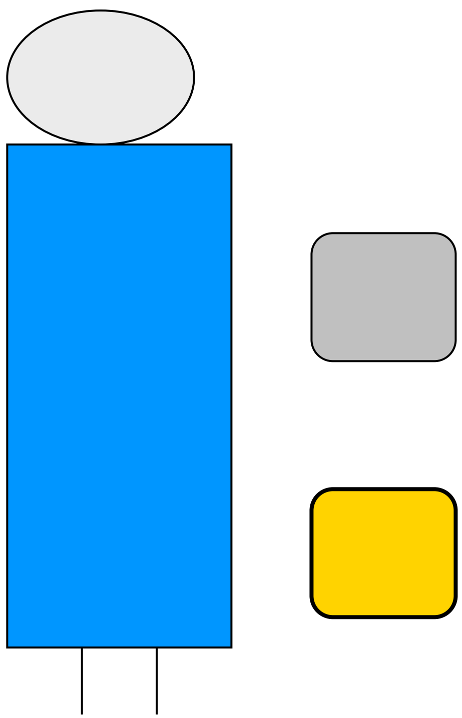
Memory



Memory



Memory

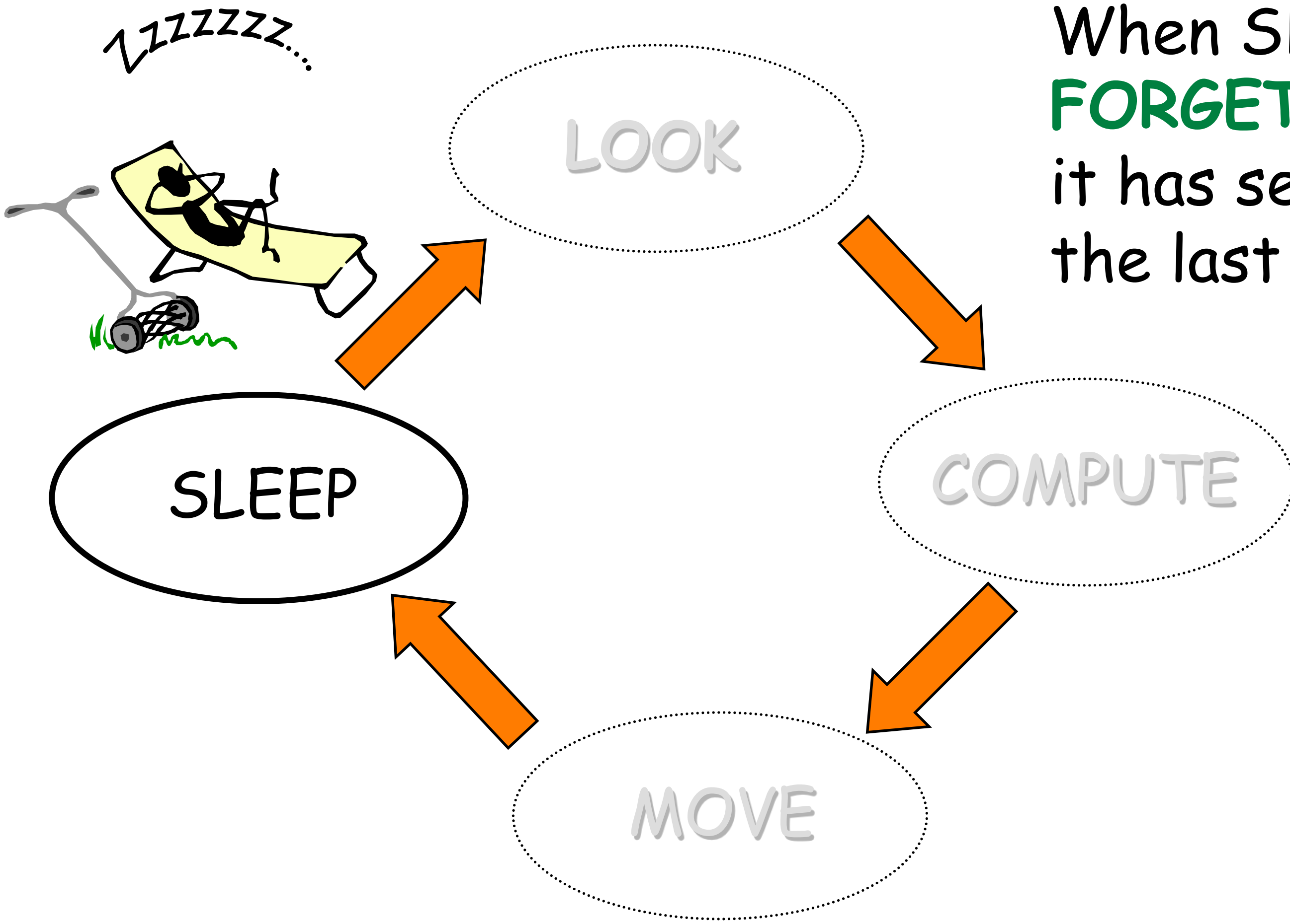


OBLIVIOUS

No memory of the past

Memory

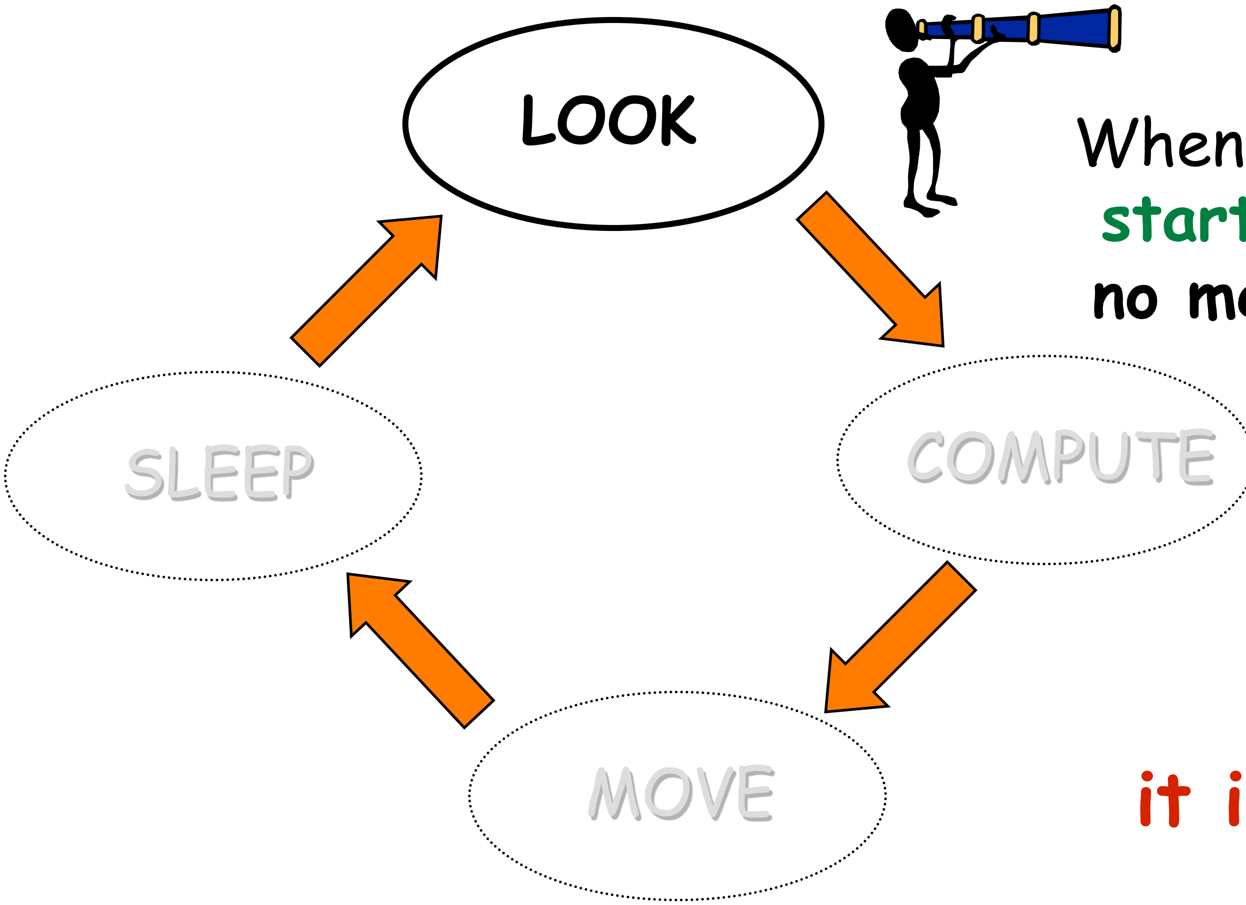
OBLIVIOUS



When Sleeping, a robot **FORGETS everything** it has seen and done in the last cycle

Memory

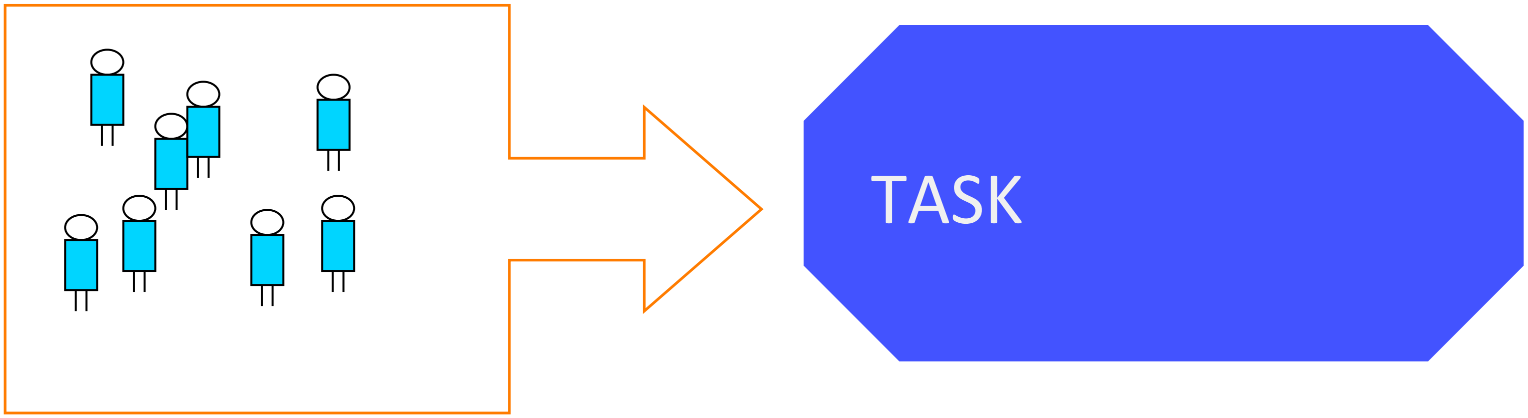
OBLIVIOUS



When Looking again, it **starts from scratch** with **no memory of the past.**

Every time it is the first time

Why Oblivious ?



Theoretical Interest: study of what capabilities are **really needed** for a cooperative team of robots to perform a given task.

Practical interest: no need of persistent memory; resilient to memory faults; self-stabilization

Crucial Factors

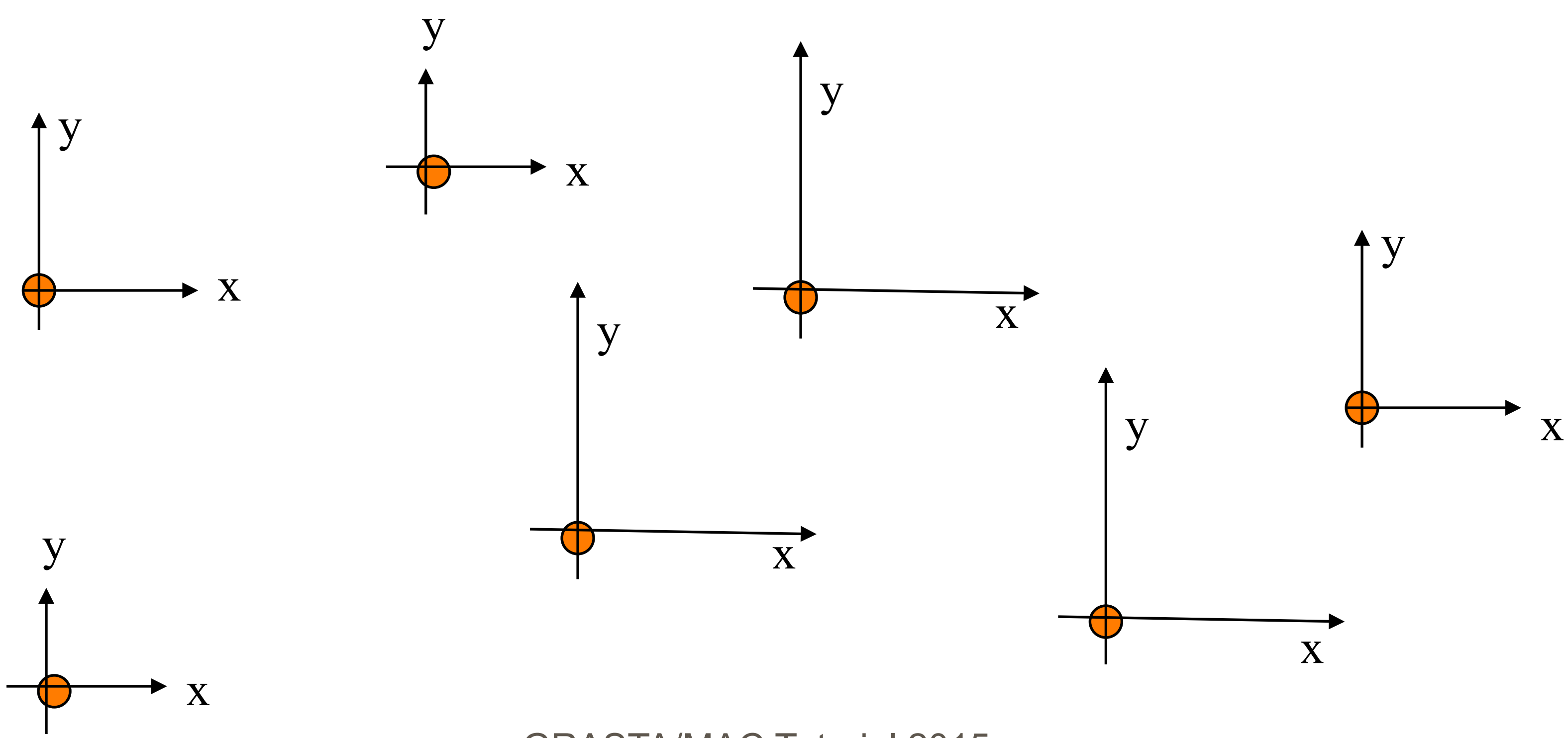
- **Visibility** Global → **Limited**
- **Memory** Persistent → **Oblivious**

Crucial Factors

- **Visibility** Global → **Limited**
- **Memory** Persistent → **Oblivious**
- **Agreement** on local coordinate systems

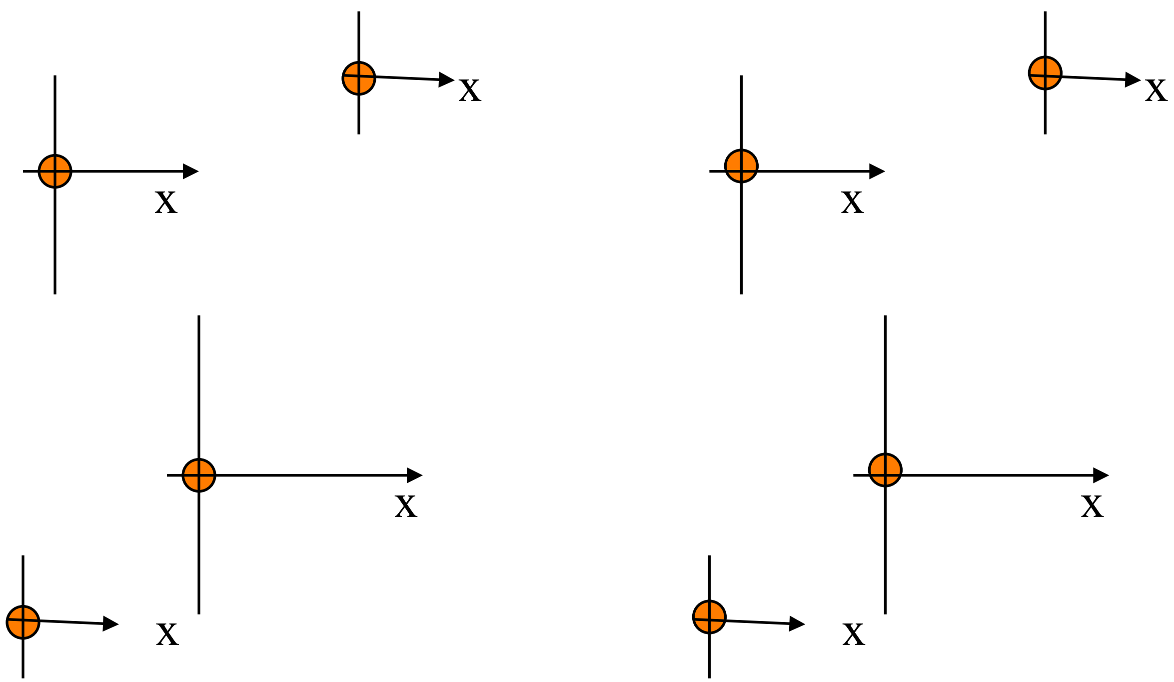
Agreement : Direction and Orientation

The robots agree on a common direction and orientation of both axes (eg compass)



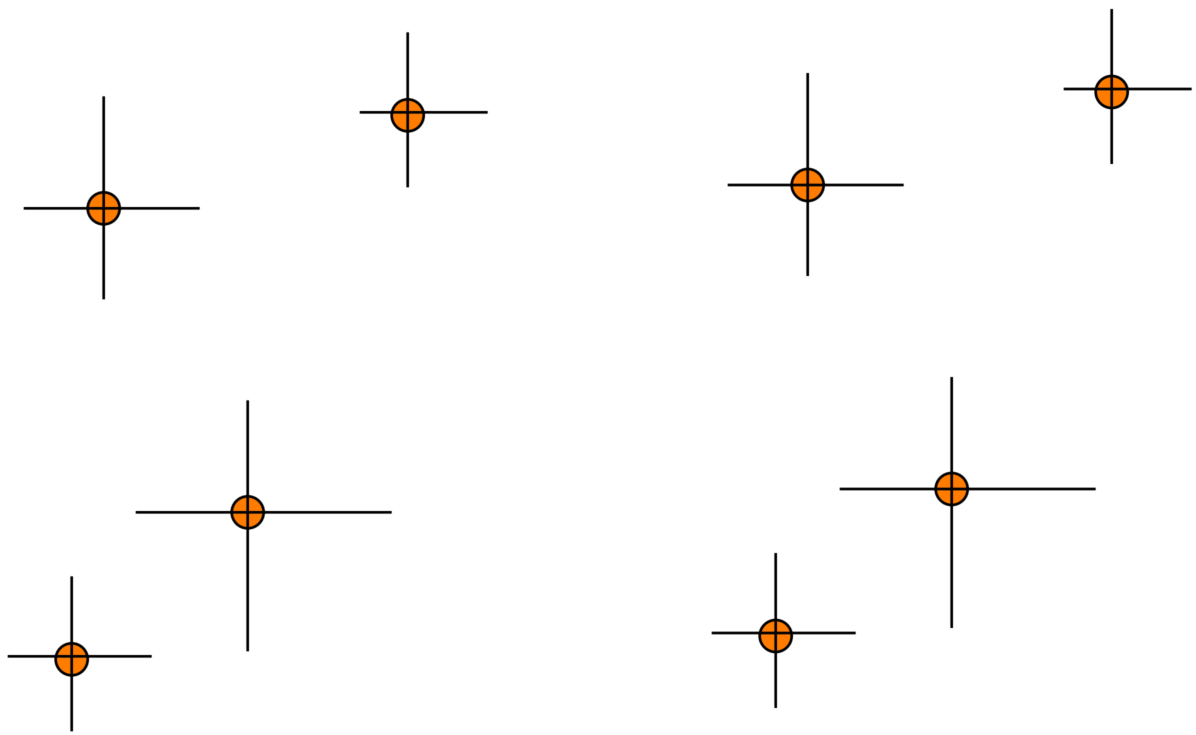
Agreement : without Chirality

The robots agree on a common direction and orientation of one axis



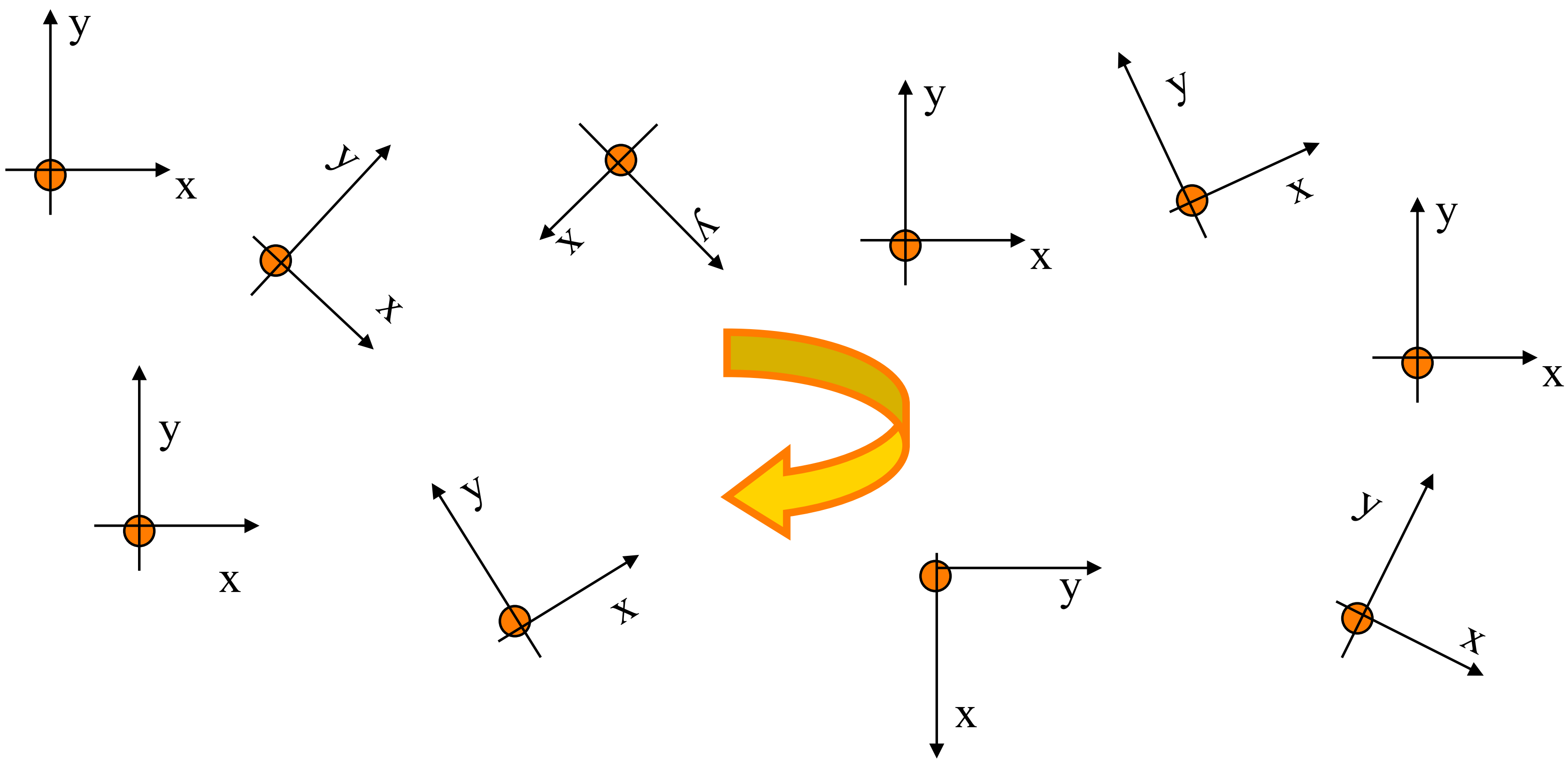
Agreement : Direction

The robots agree on direction of both axes

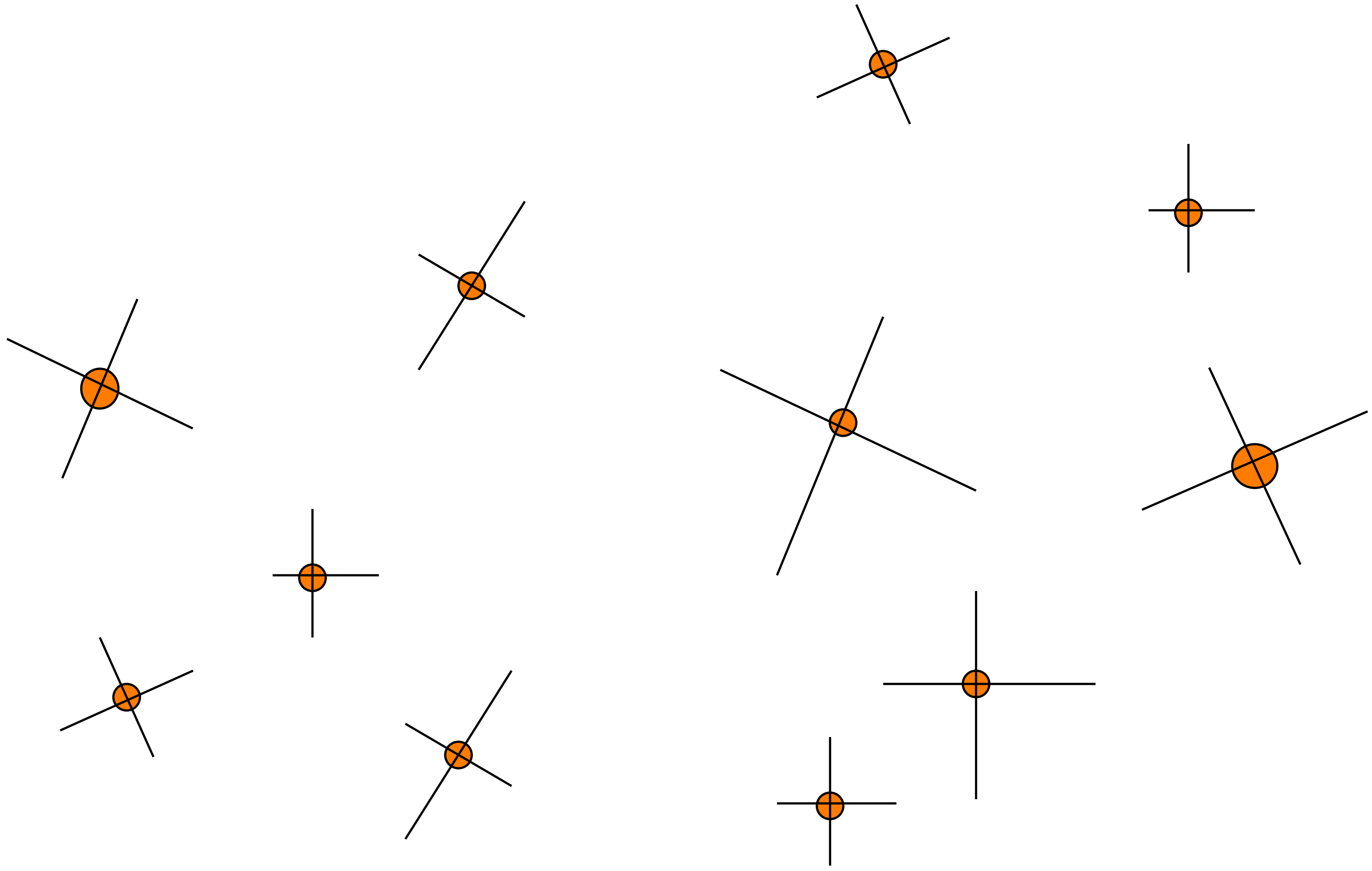


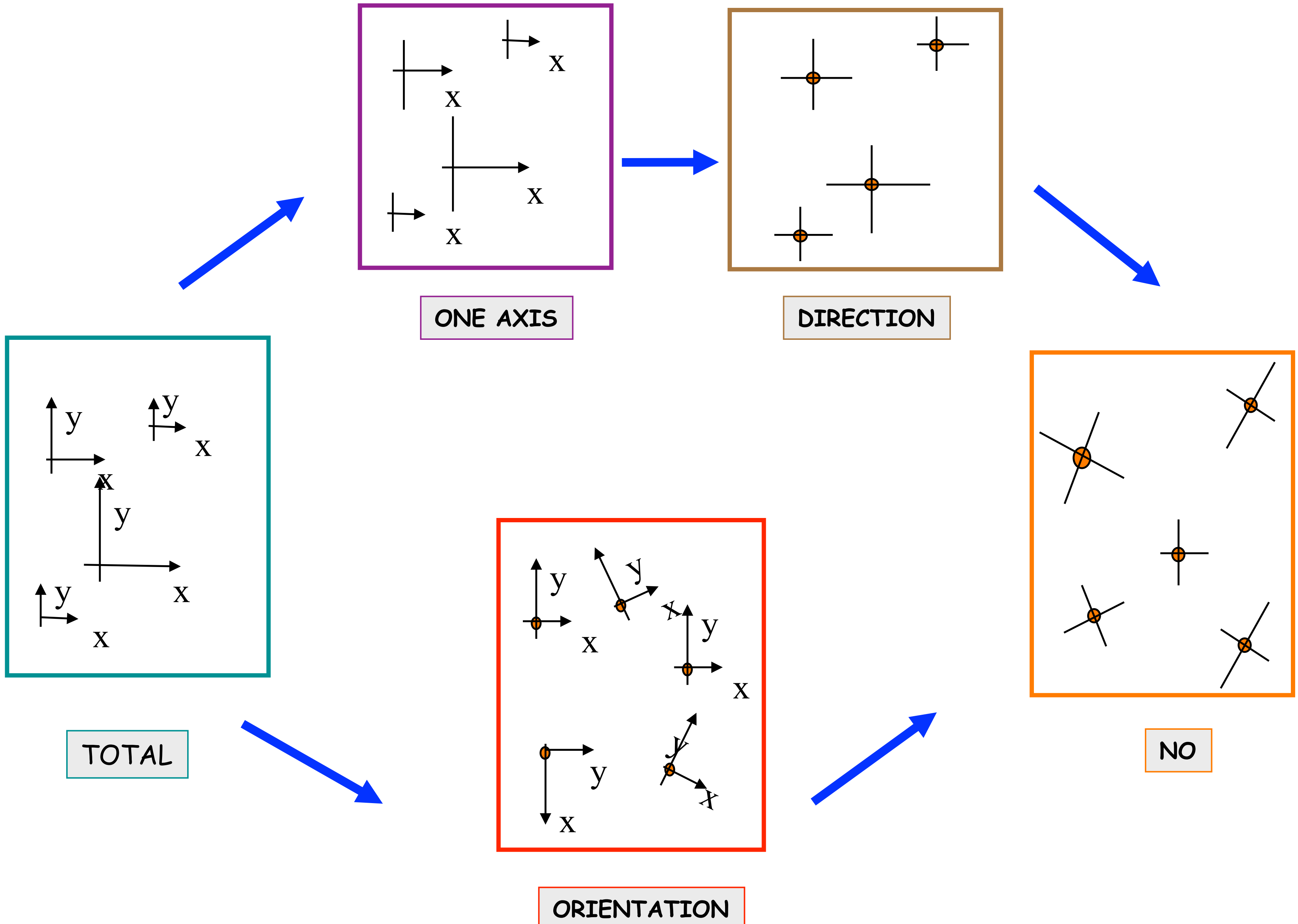
Agreement 4: Chirality

The robots agree on **circular orientation** of plane



NO Agreement

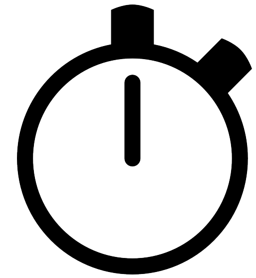



ONE AXIS
DIRECTION
TOTAL
NO
ORIENTATION

Crucial Factors

- Visibility Global → **Limited**
- Memory Persistent → **Oblivious**
- Agreement Total → **No**
- Time/Synchronization

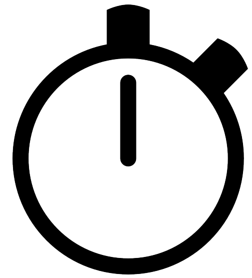
Time / Synchronization



There are three basic models

- **Fully synchronous** (FSYNC)
- **Semi synchronous** (SSYNC)
- **Asynchronous** (ASYNC)

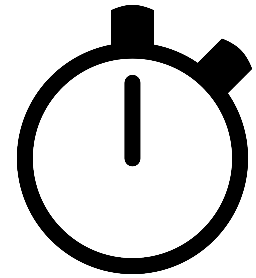
Synchronous Systems



- Fully synchronous (FSYNC)

- Semi synchronous (SSYNC)

Synchronous Systems

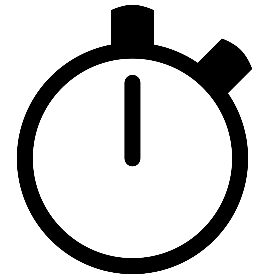


- **Fully synchronous** (FSYNC)
- **Semi synchronous** (SSYNC)

There is a **global clock tick** reaching all robots simultaneously

At each clock tick every robot is either active or inactive; active robots perform their cycle **atomically**.

Synchronous Systems



- **Fully synchronous** (FSYNC)

  all robots are active

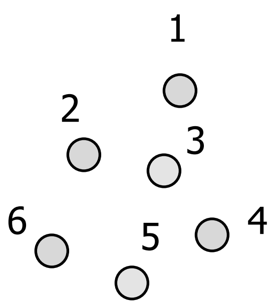
- **Semi synchronous** (SSYNC)

  subset of robots are active

There is a **global clock tick** reaching all robots simultaneously

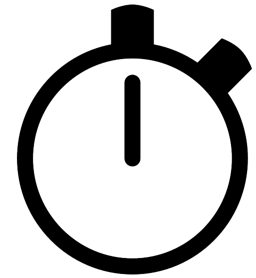
At each clock tick every robot is either active or inactive; active robots perform their cycle **atomically**.

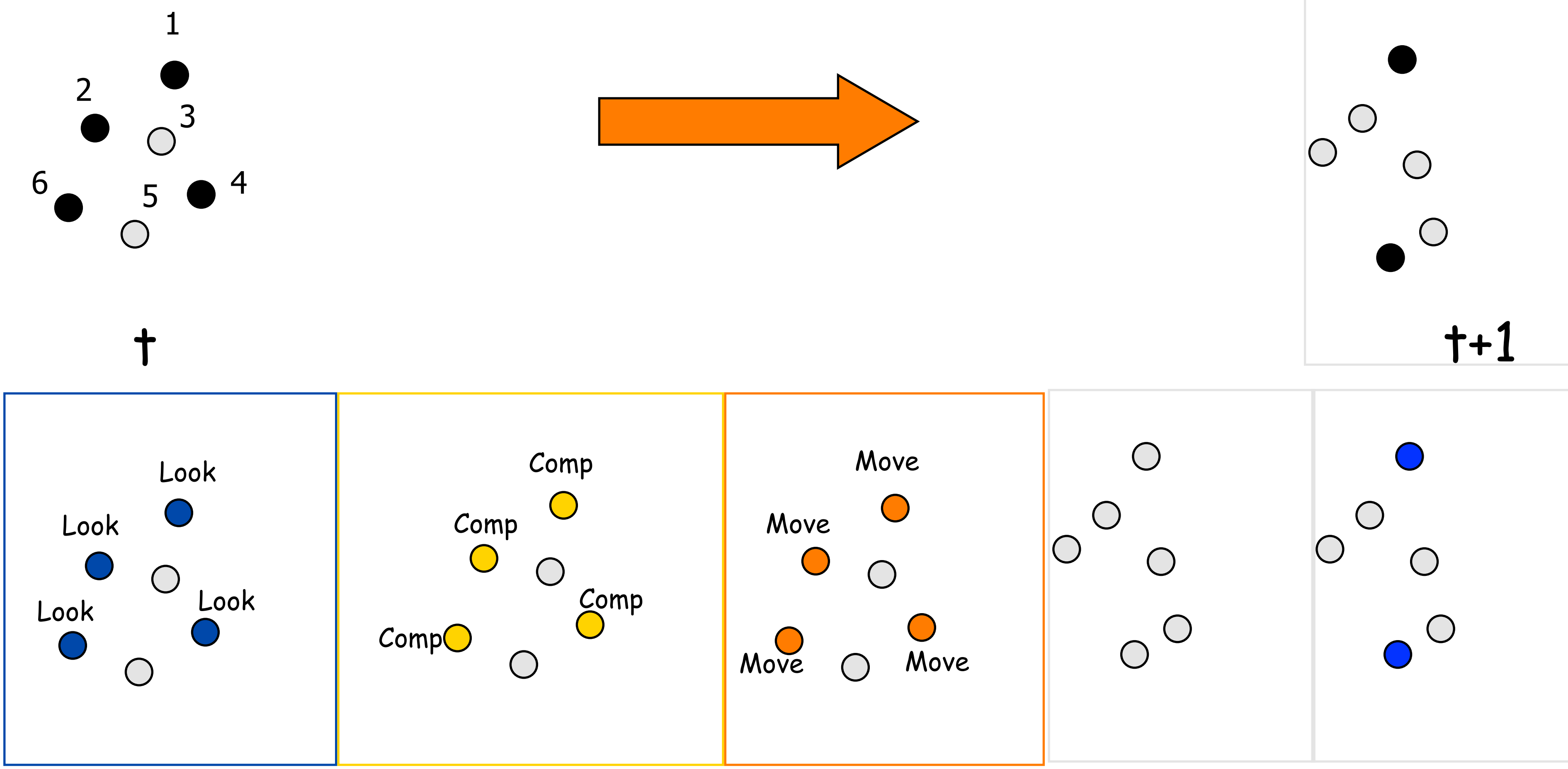
Synchronous Systems



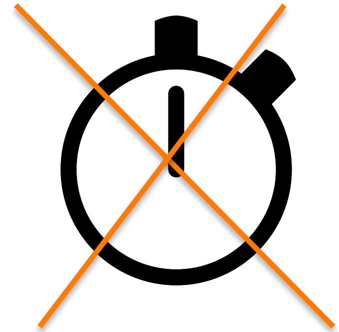
t

Synchronous Systems



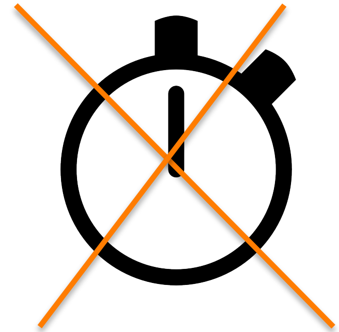


Asynchronous Systems (ASYNC)

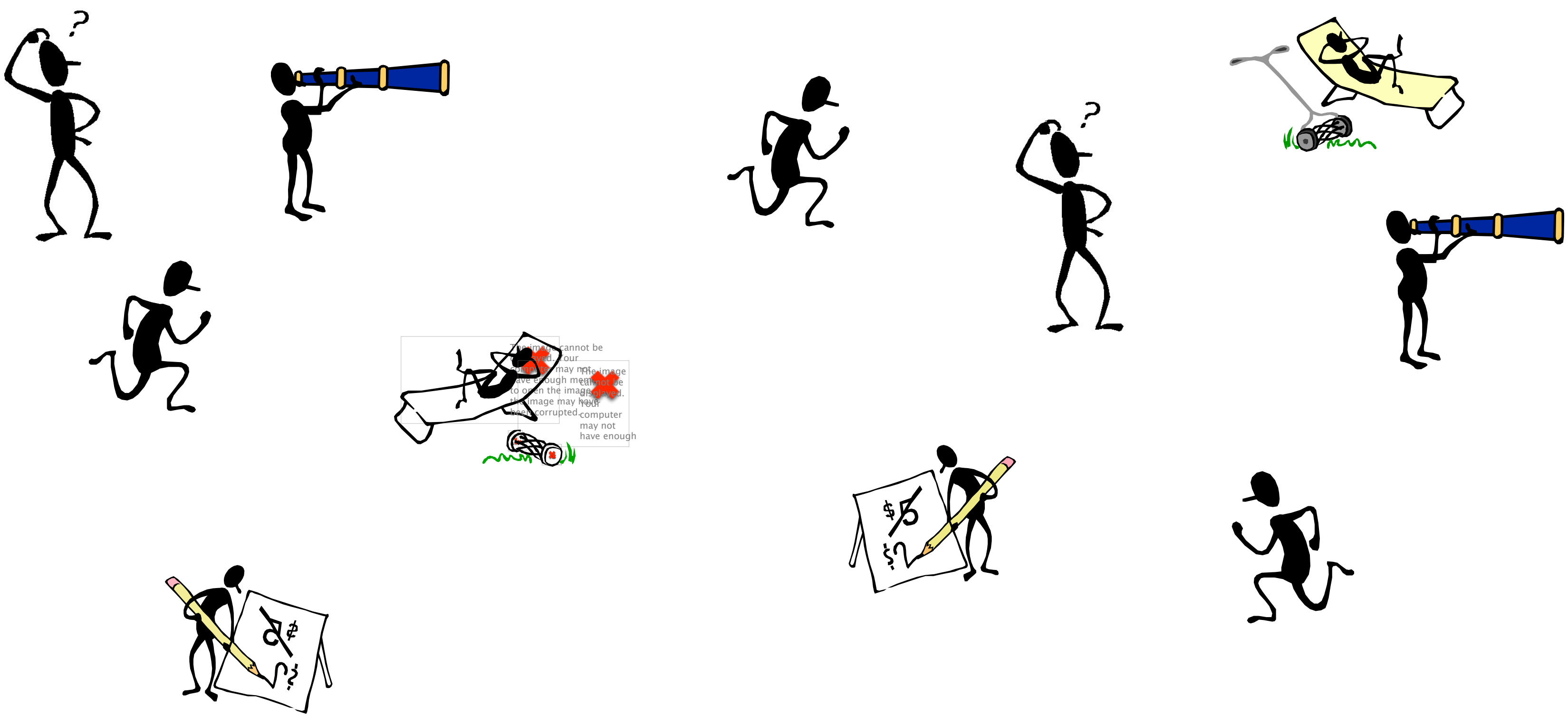


--- There is **NO global clock** and robots do not have a common notion of time

Asynchronous Systems (ASYNC)



--- There is NO global clock and robots do not have a common notion of time



Asynchronous Systems (ASYNC)

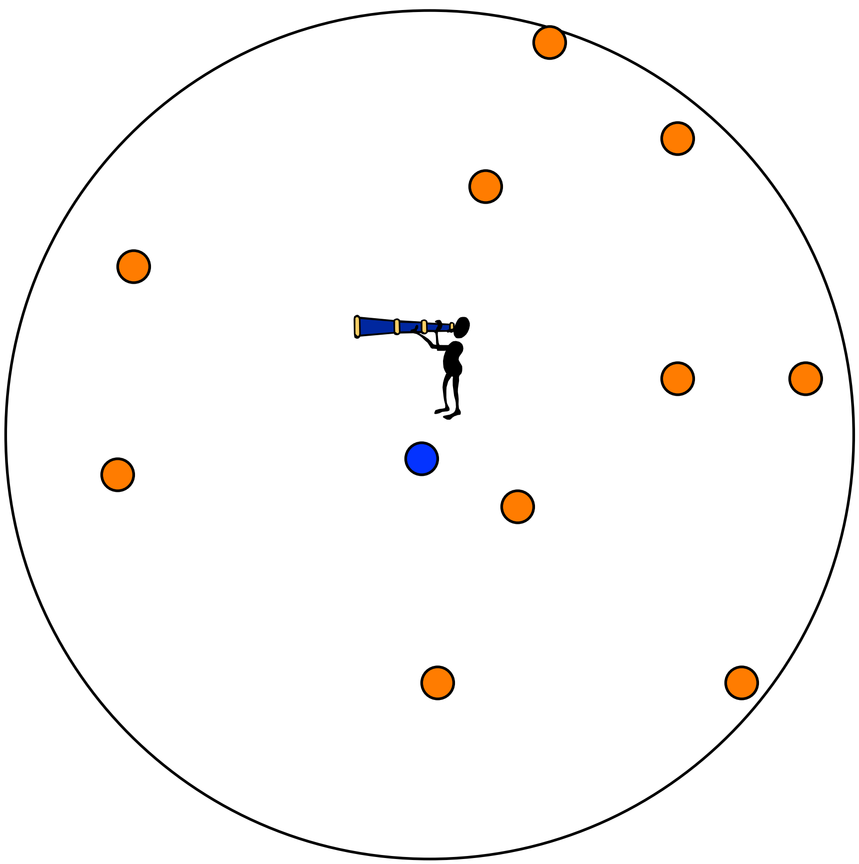
-There is **NO** global clock and robots do not have a common notion of time

- Each robot becomes active at **unpredictable** time instants

- Each computation and movement takes a finite but **unpredictable** amount of time

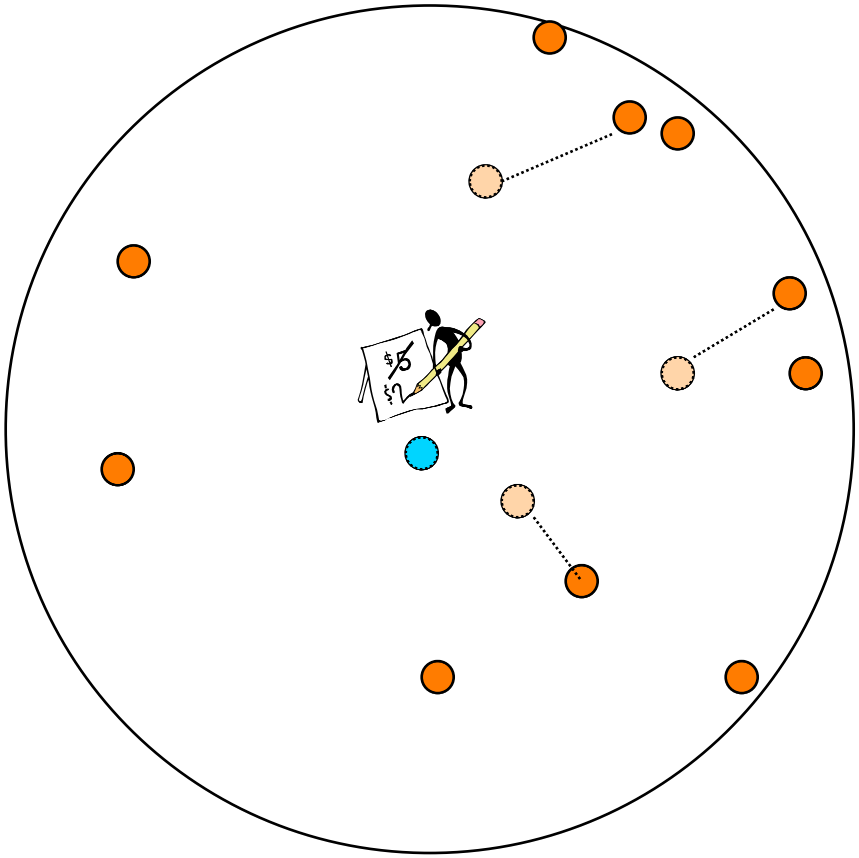
Only the Looking phase is atomic

Asynchronous Systems (ASYNC)



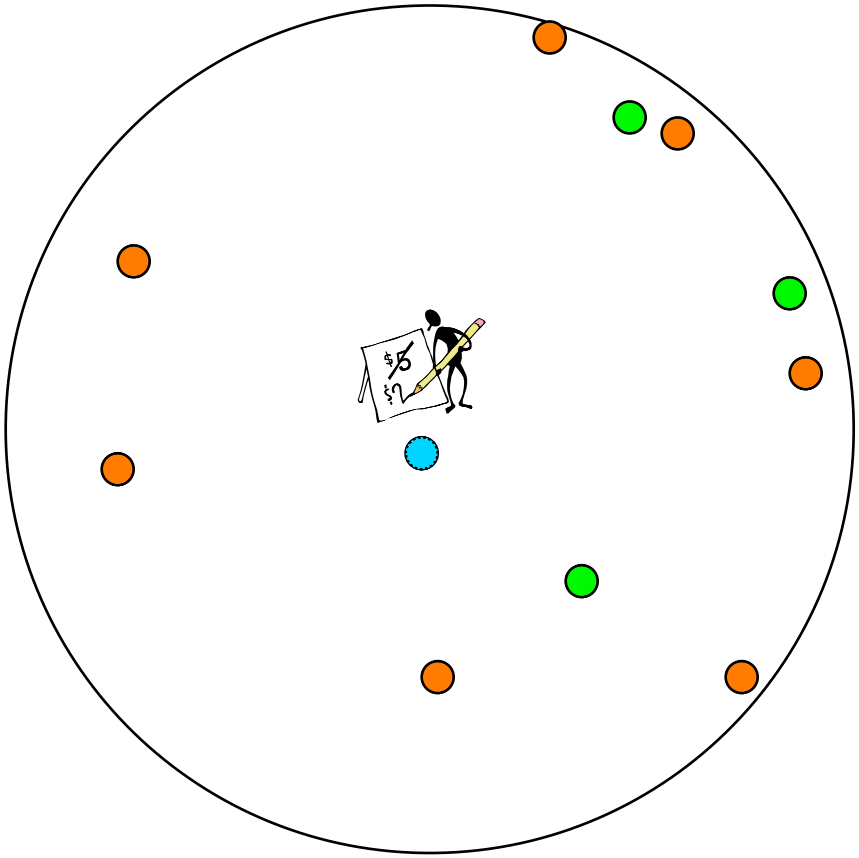
Asynchronous Systems (ASYNC)

COMPUTE



Asynchronous Systems (ASYNC)

COMPUTE

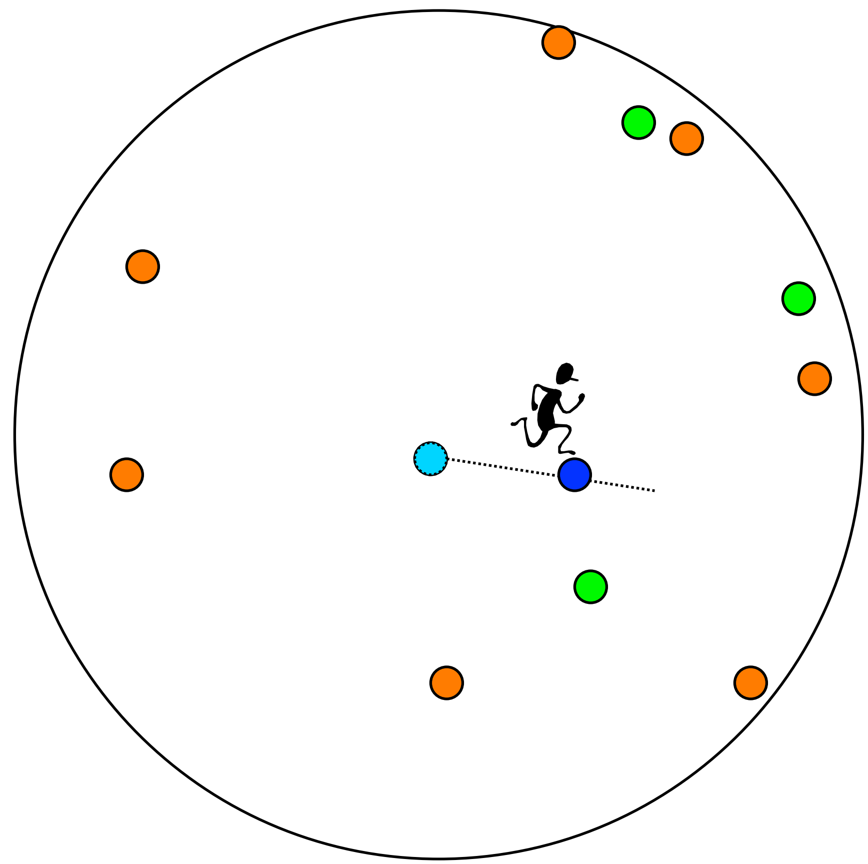


Computation based on obsolete information

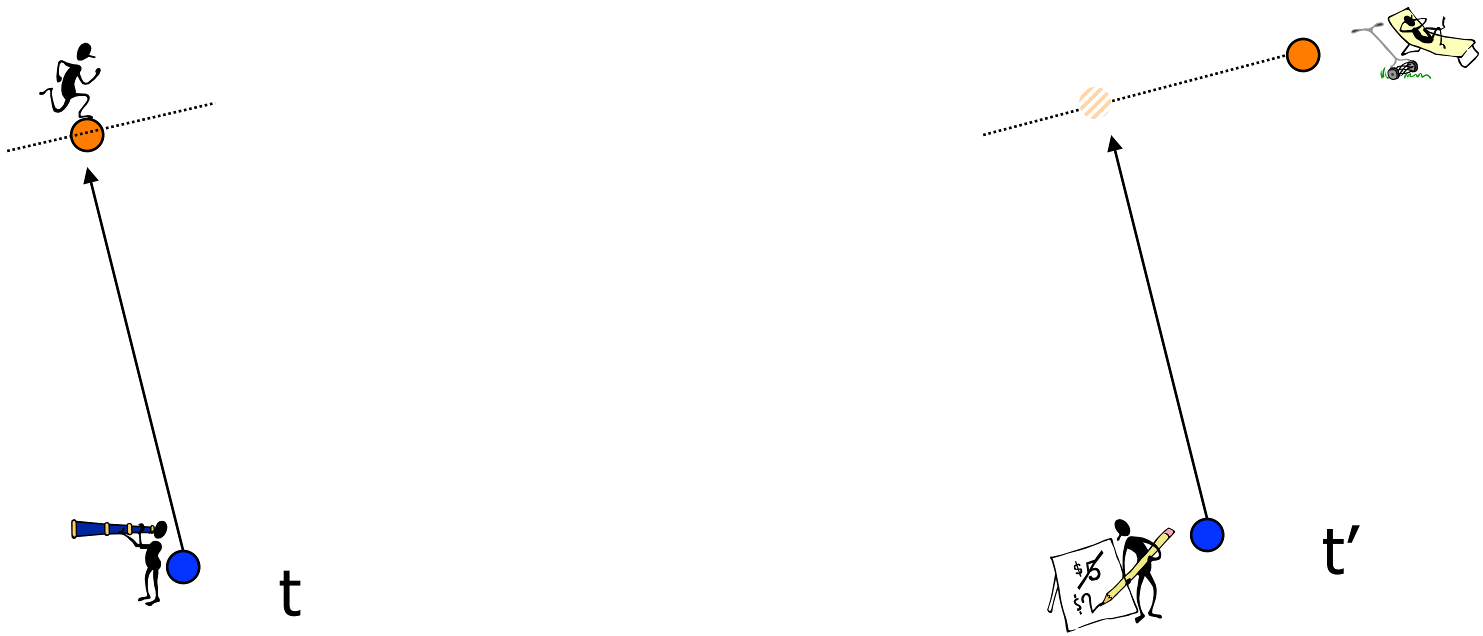
Asynchronous Systems (ASYNC)

MOVE

Movement based on obsolete information



Asynchronous Systems (ASYNC)



A robot could see other robots while they move !

Crucial Factors

- Visibility Global → **Limited**
- Memory Persistent → **Oblivious**
- Agreement Total → **No**
- Time/Synchronization FSYNCH → ***ASYNCH***

Algorithmic Investigations

- Suzuki,Yamashita (SIROCCO '98, *SIAM J* Computing 00)
- Ando,Oasa,Suzuki,Yamashita (*Robotics & Automat.* 99)
- Flocchini,Prencipe, Santoro,Widmayer (ISAAC'99, TCS 10)
- Flocchini,Prencipe, Santoro,Widmayer (STACS'01, TCS 05)
- Flocchini,Prencipe, Santoro,Widmayer (SIROCCO 01)
- Cielibak,Prencipe (SIROCCO 02)
- Defago, Konegaya (PMC '02, TCS 2011)
- Gervasi, Prencipe (*DAM* '03)
- Cielibak,Flocchini,Prencipe,Santoro (ICALP'03)
- Souissi,Defago,Katayama (JSF'04)(TCS 09)
- Agmon, Peleg (SODA '04, SIAM J. Computing)
- Chatzigiannakis,Markou,Nikoletseas (WEA 04)
- Cielibak (LATIN 04)

.....

Algorithmic Investigations

Flocchini, Prencipe, Santoro (ALGOSENSOR 06, TCS)
Cohen, Peleg (ESA04) (STACS 06) (SIROCCO 06) (SIAM J. Computing)
Dieudonne, Labbane, Petit (SSS06) (ACM Transactions on Autonomous and Adaptative System, 2008)
Katayama et al (SIROCCO07)
Dieudonne, Petit (SIROCCO 07, IPL 07)
Czyzowicz, Gasieniec, Pelc (OPODIS 00) (*TCS* 09)
Cohen,Peleg (SODA 04) (SIROCCO 04)
Prencipe (SIROCCO 05) (*TOCS* 05)
Izumi, Katayama, Inuzuka, Wada (DISC 07)
Dieudonne, Petit (DISC 09)
Das, Flocchini, Santoro, Yamashita (PODC 2010)
Kamei, Lamani, Ooshita, Tixeuil (SIROCCO '11)

....

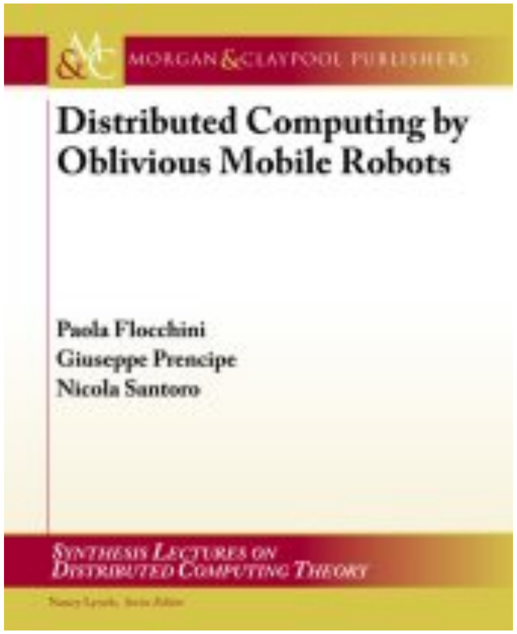
Algorithmic Investigations

Cielibak,Flocchini,Prencipe,Santoro (*SIAM J. Comp*, '12)
Izumi,Souissi,Katayama,,Defago,Wada,Yamashita (*SIAM J. Comp.*, '12)
Yamamoto, Izumi, Katayama, Inuzuka,Wada (*Theo.Comp.Sci.'* 12)
Dieudonne, Petit (*Theo.Comp.Sci.* '12)
Das.Flocchini, Prencipe, Santoro,Yamashita (ICDCS '12)
Dieudonné, F Levé, F Petit, V Villain (*Theo.Comp.Sci.'* 13)
Izumi,Kamei, Ooshita (*Parallel Distributed Systems* '13)
Das, Flocchini, Prencipe, Santoro (FUN'14)
Di Stefano, Navarra (*SSS*'14)
Lukovszki, M. auf der Heide (OPODIS '14)
Flocchini, Prencipe, Santoro, Viglietta (OPODIS '14)
Gan Chaudhuria,Mukhopadhyaya (*J. Discrete Alg.* '15)
Fujinaga, Yamauchi, Ono, Kijima, Yamashita (*SIAM J. Comp*, '15)
Das, Flocchini, Santoro, Yamashita (*Distributed Computing* '15)

AND MANY MANY MORE …

Algorithmic Investigations

P. Flocchini, G. Prencipe, N. Santoro
Distributed Computing by Oblivious Mobile Robots
Morgan & Claypool, 2012



PATTERN FORMATION

global visibility
asynchronous
oblivious

Pattern Formation

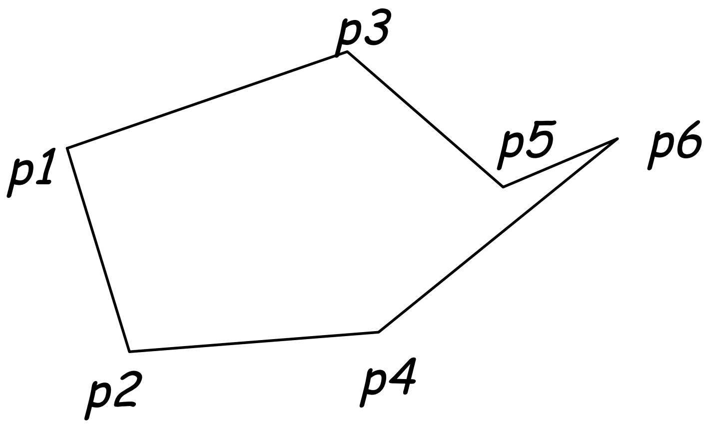
Pattern ***P*** a set of (distinct) points given in input.

Configuration ***C*** a set of (distinct) points where the robots are.

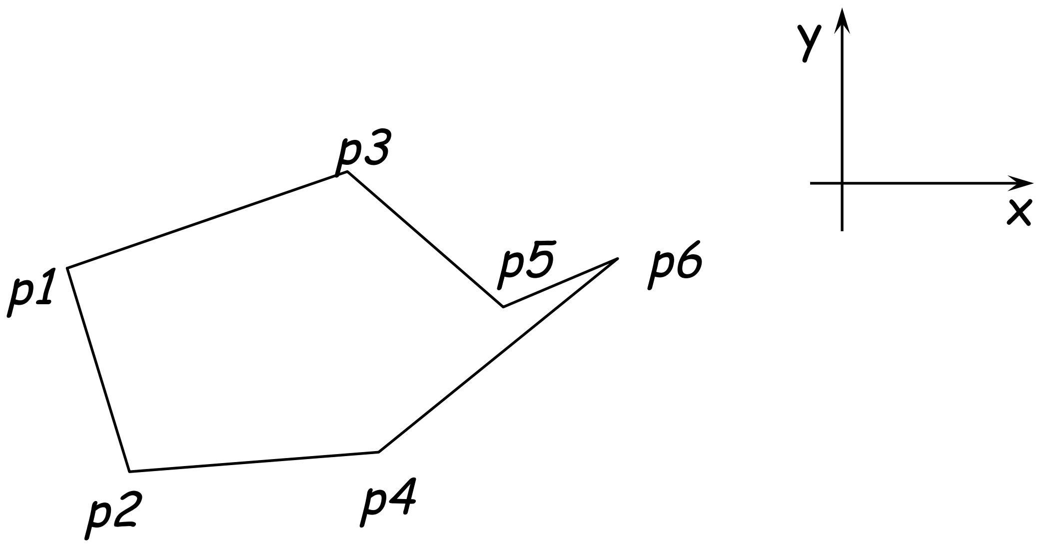
Starting from **any** arbitrary initial configurations must form a configuration ***C*** equal to ***P***

(*translation*, *scaling* or *rotation*).

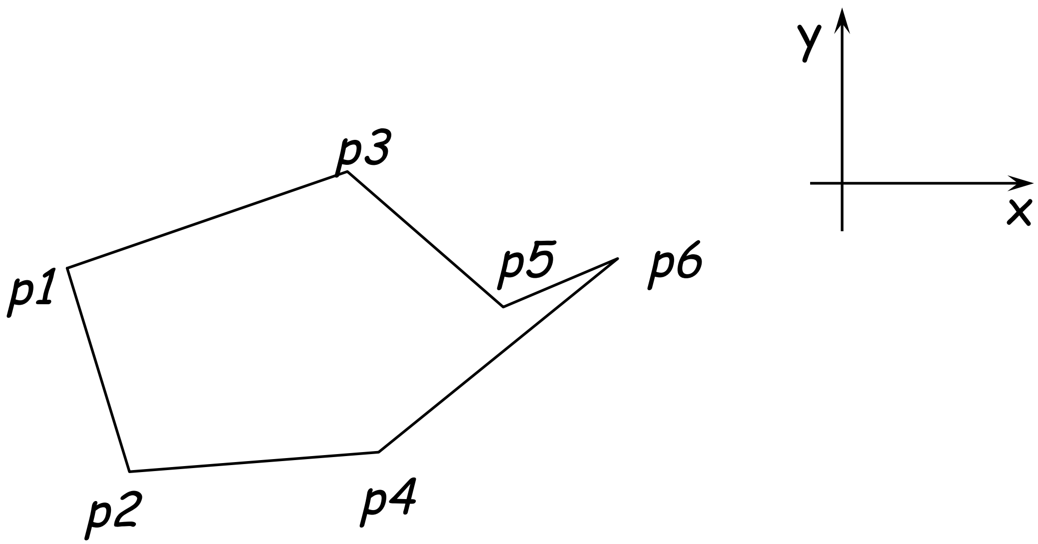
Pattern Formation

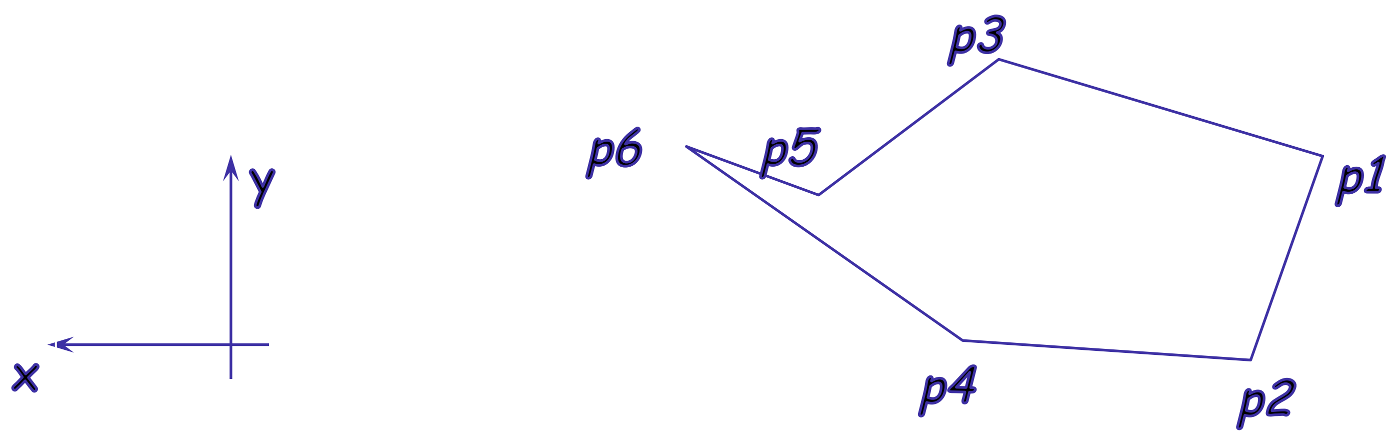


Pattern Formation

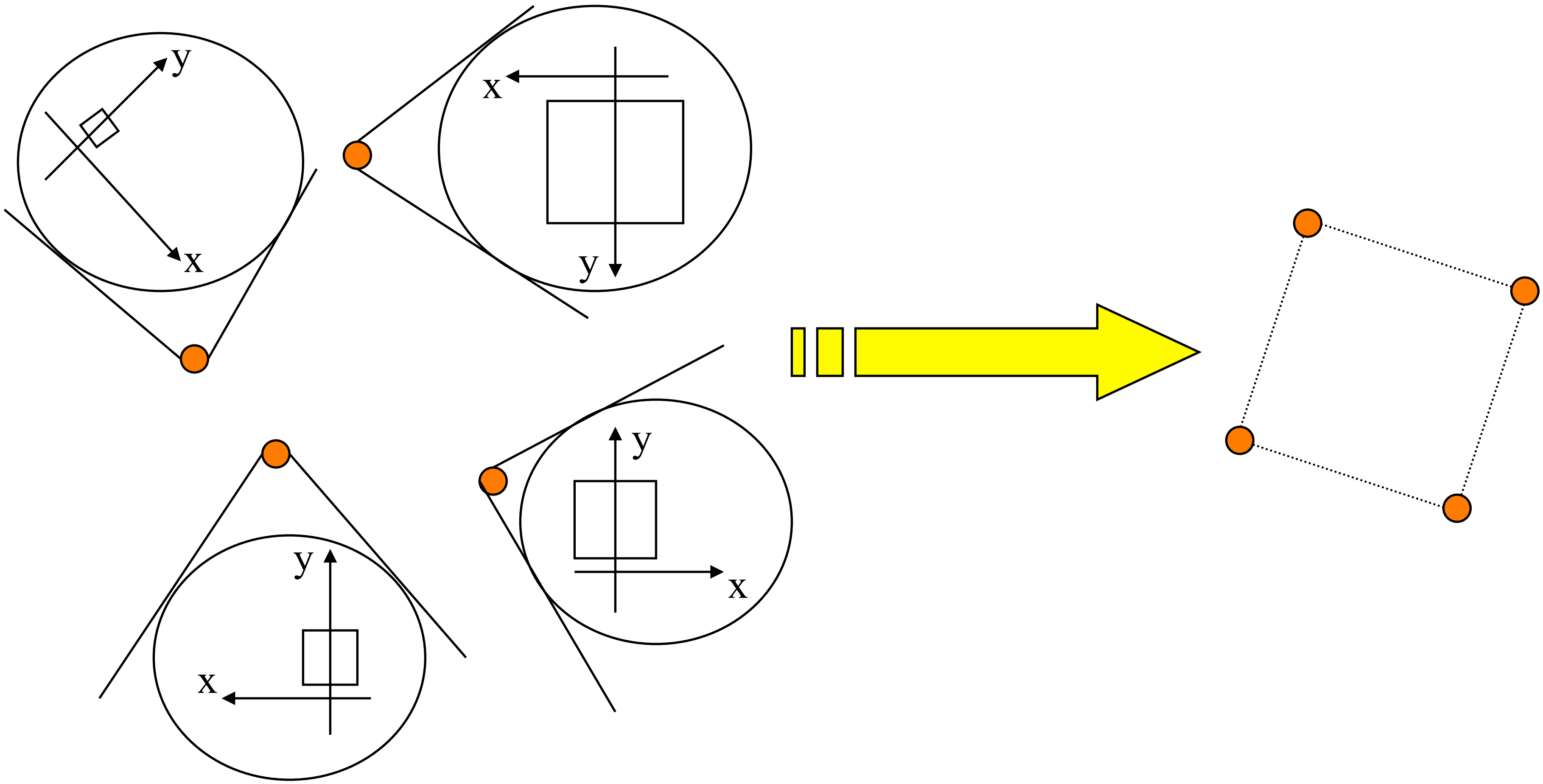


Pattern Formation





Pattern Formation



Pattern Formation

what patters are formable from a given configuration ?

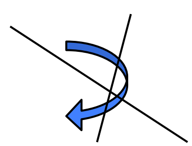
Suzuki, Yamashita. *SIAM J. Computing*, '99 **SSYNC+chirality**

Pattern Formation

what patters are formable from a given configuration ?

Suzuki, Yamashita. *SIAM J. Computing*, '99 — **SSYNC+chirality**

Fujinaga, Yamauchi, Ono, Kijima, Yamashita. *SIAM J. Comp*, '15 — **ASYNC+chirality**



Without chirality ? Still open

Pattern Formation

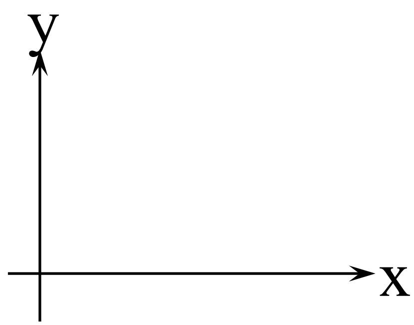
what patters are formable from every configuration with a given coordinate agreement ?

Pattern Formation

what patters are formable from every configuration with a given coordinate agreement ?

Flocchini,Prencipe,Santoro,Widmayer. *Theor. Comp. Sci.*, '08. **ASYNC**

Pattern Formation

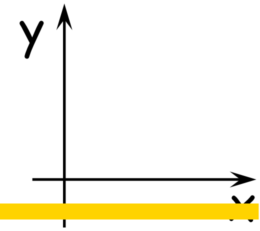


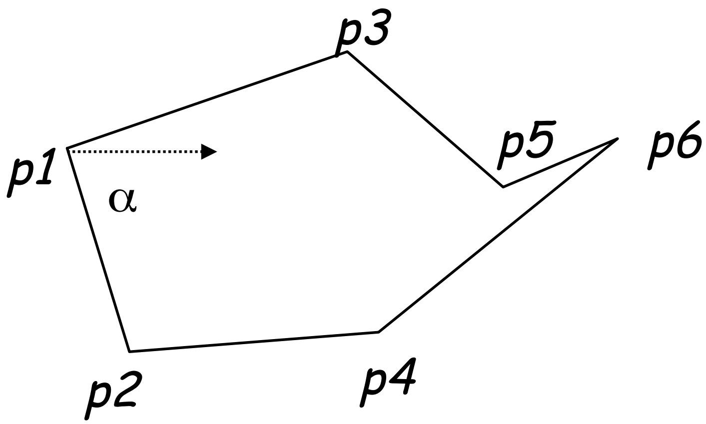
Agreement on Coordinate System
(e.g., compass, GPS)

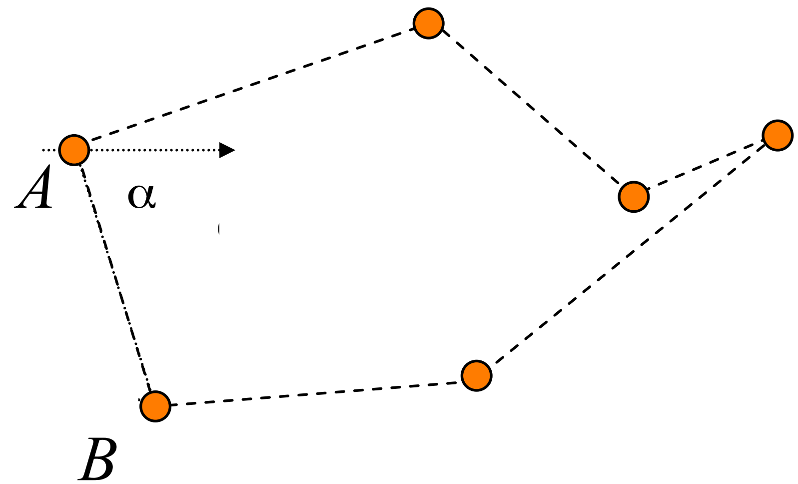
The robots can form ***any*** pattern starting from ***any*** configuration

even if the robots are **asynchronous** and **oblivious**

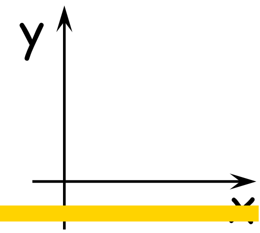
Pattern Formation





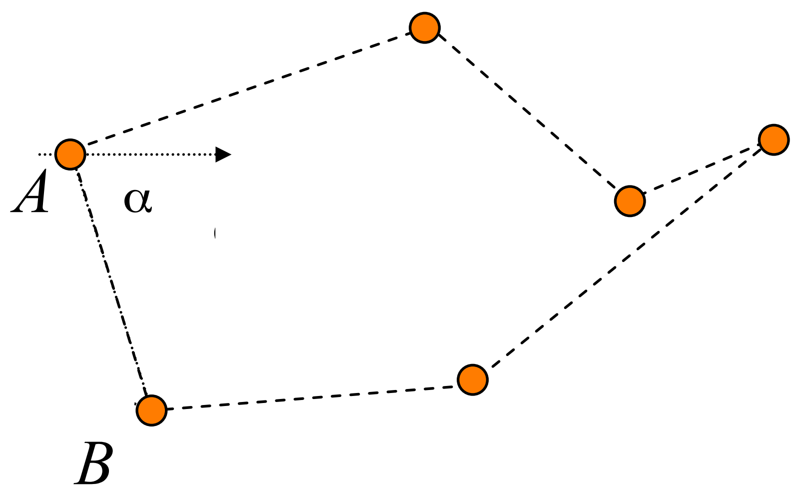


Pattern Formation



If I am the leftmost ….

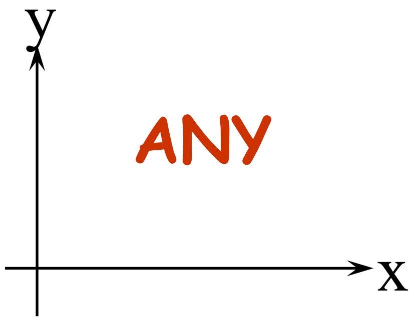
If I am the second leftmost ….



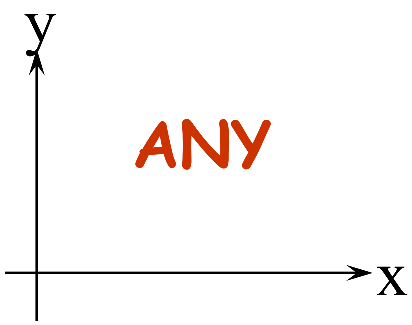
If I am not the leftmost and the angle between the leftmost and the second leftmost is ≠ α: I DON'T MOVE

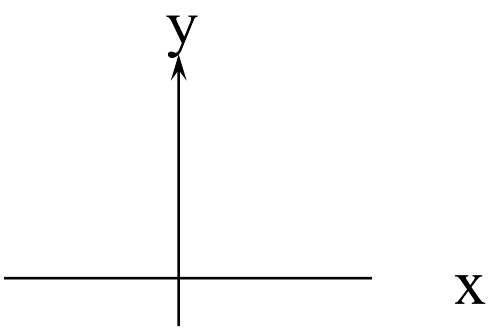
If I am not the leftmost and the angle between the leftmost and the second leftmost is = α: …..

Pattern Formation



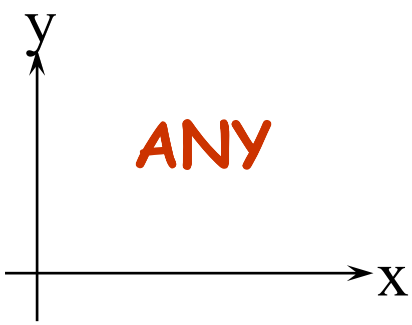
Pattern Formation



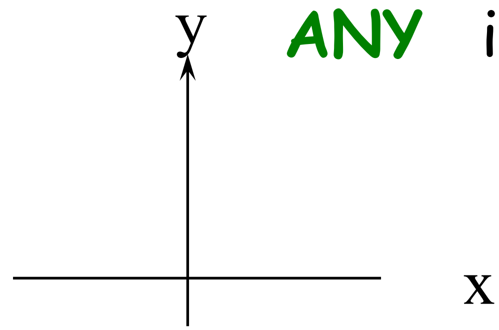


no chirality

Pattern Formation



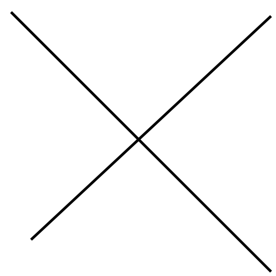
ANY if n odd



no chirality

even if asynchronous and **oblivious**

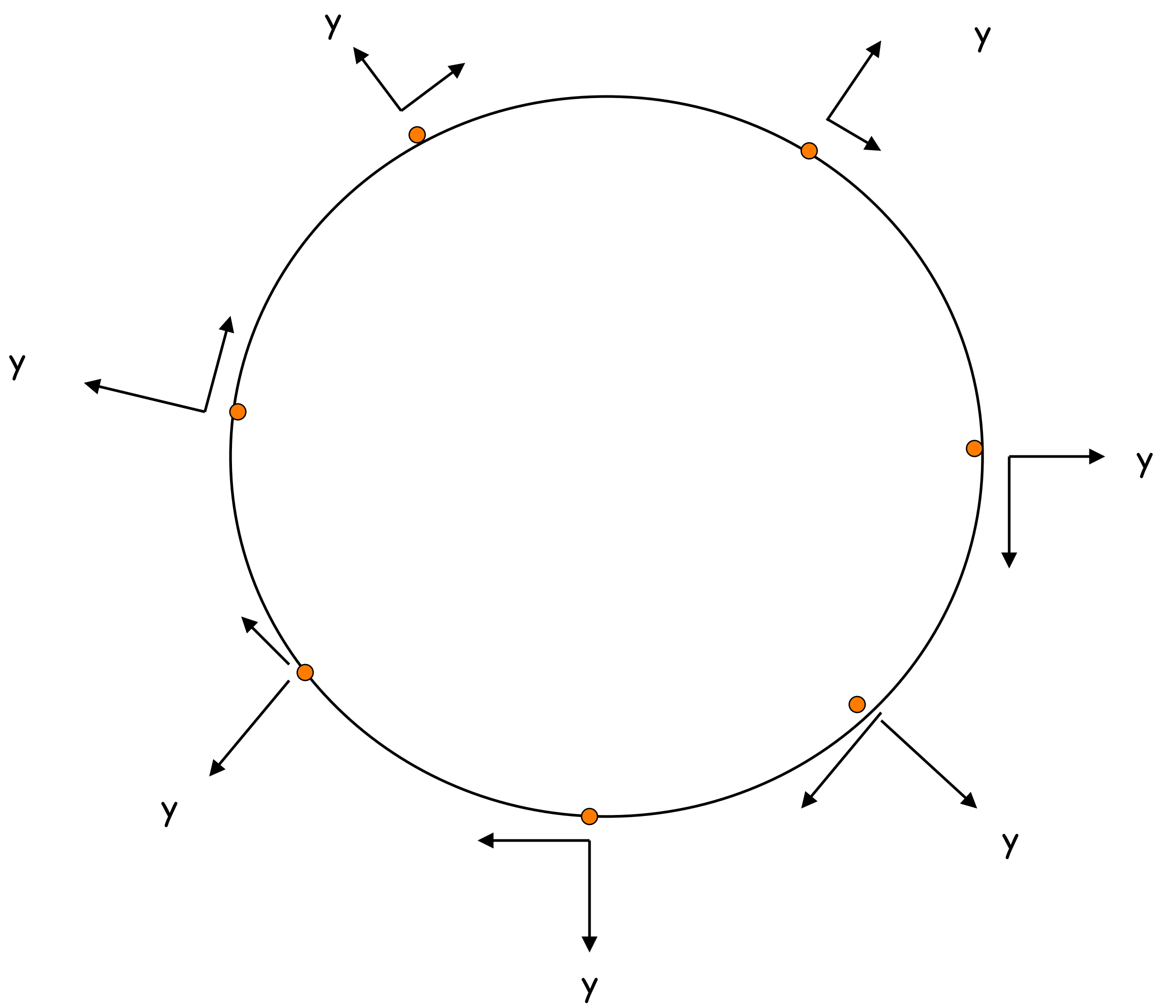
Pattern Formation



no agreement

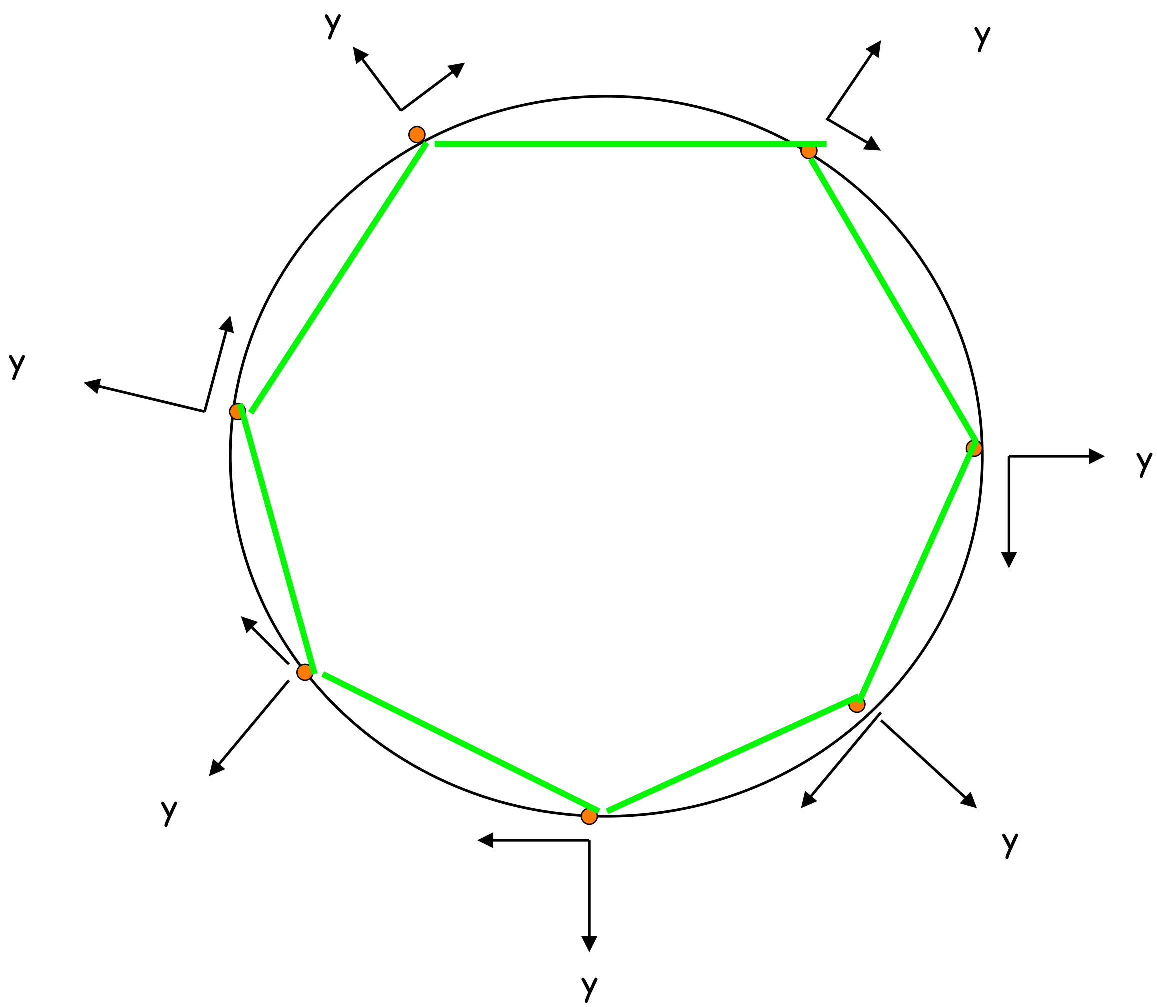
What patterns can be formed
regardless of initial configuration ?

Consider this initial configuration

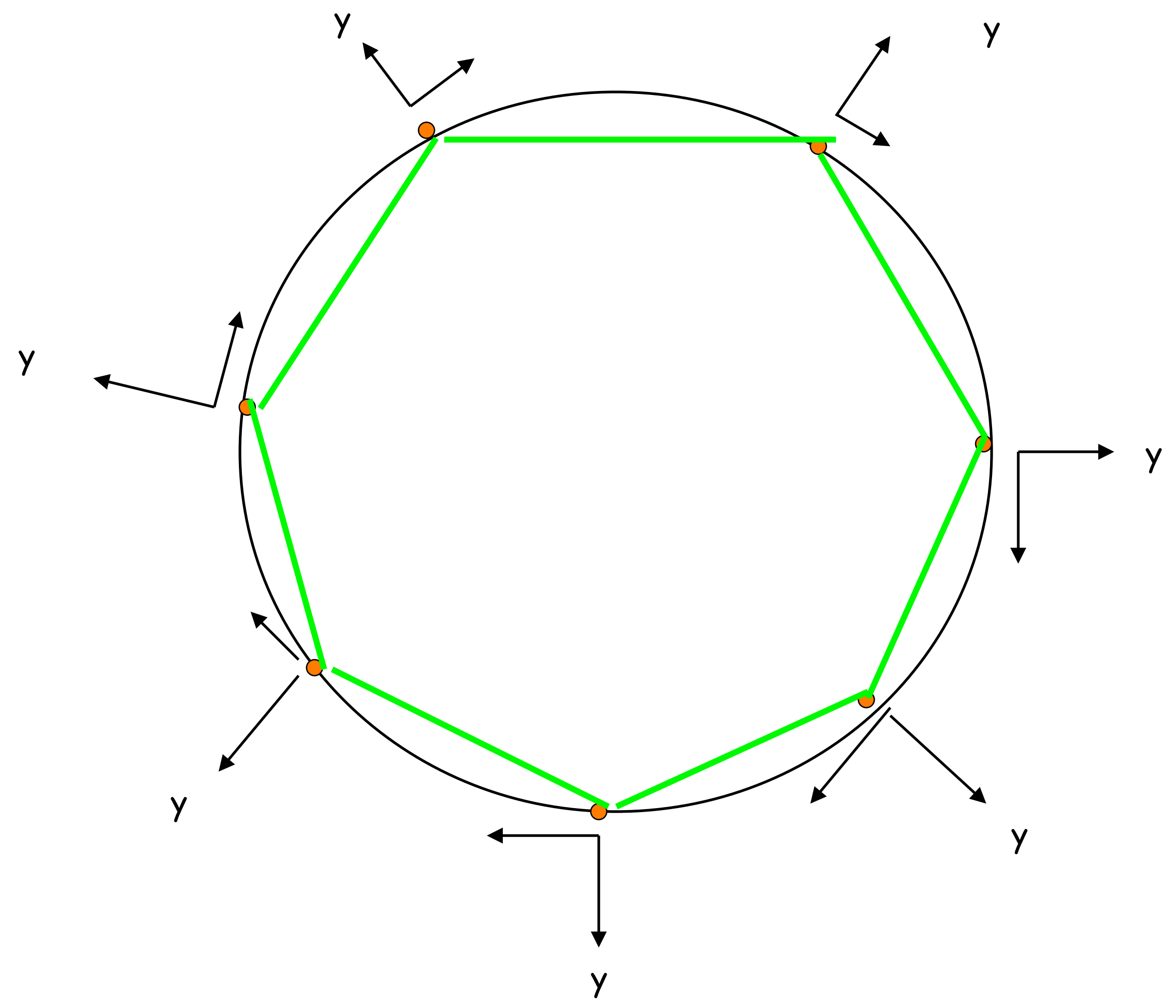


Consider this initial configuration

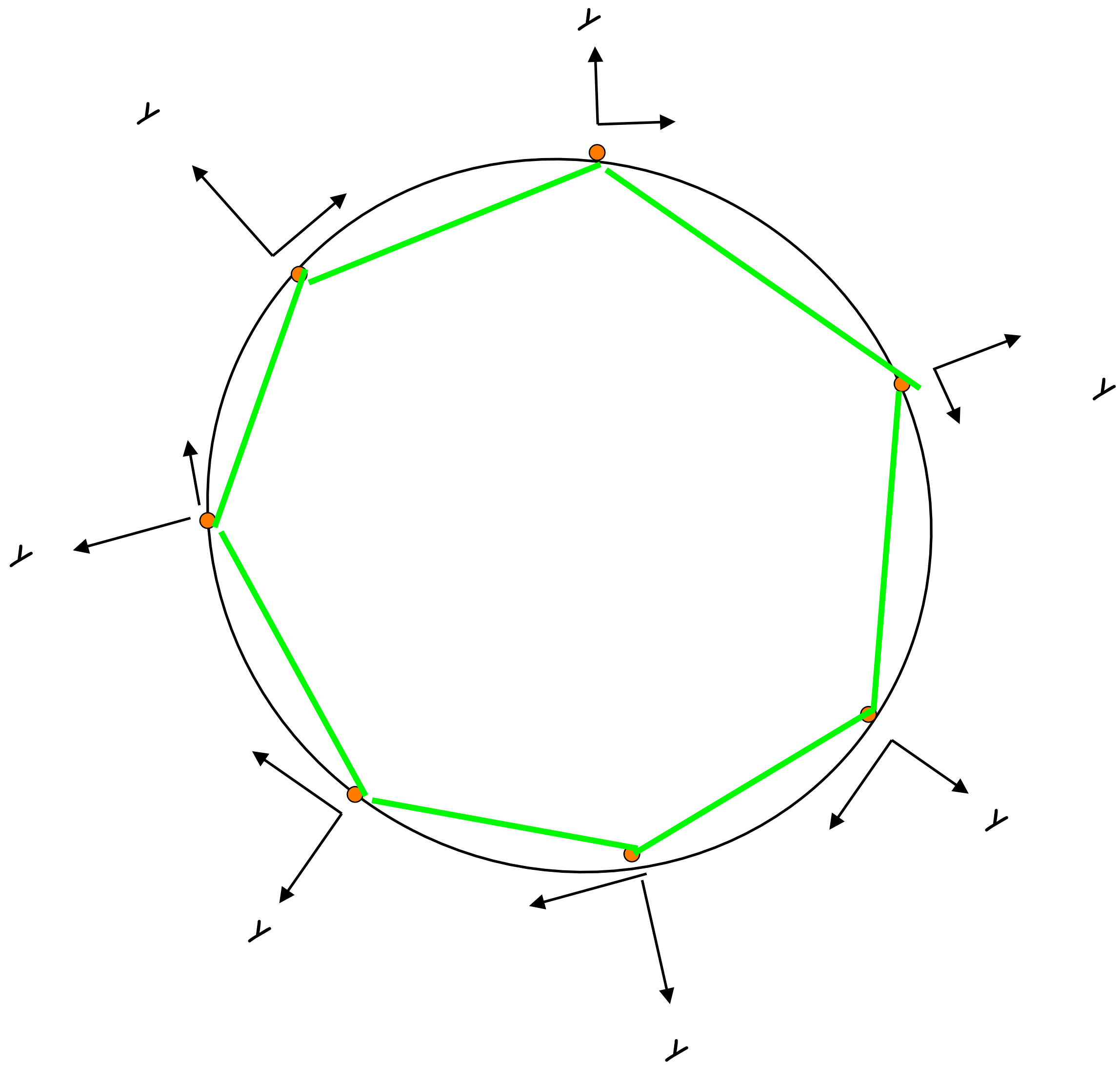
n-GON



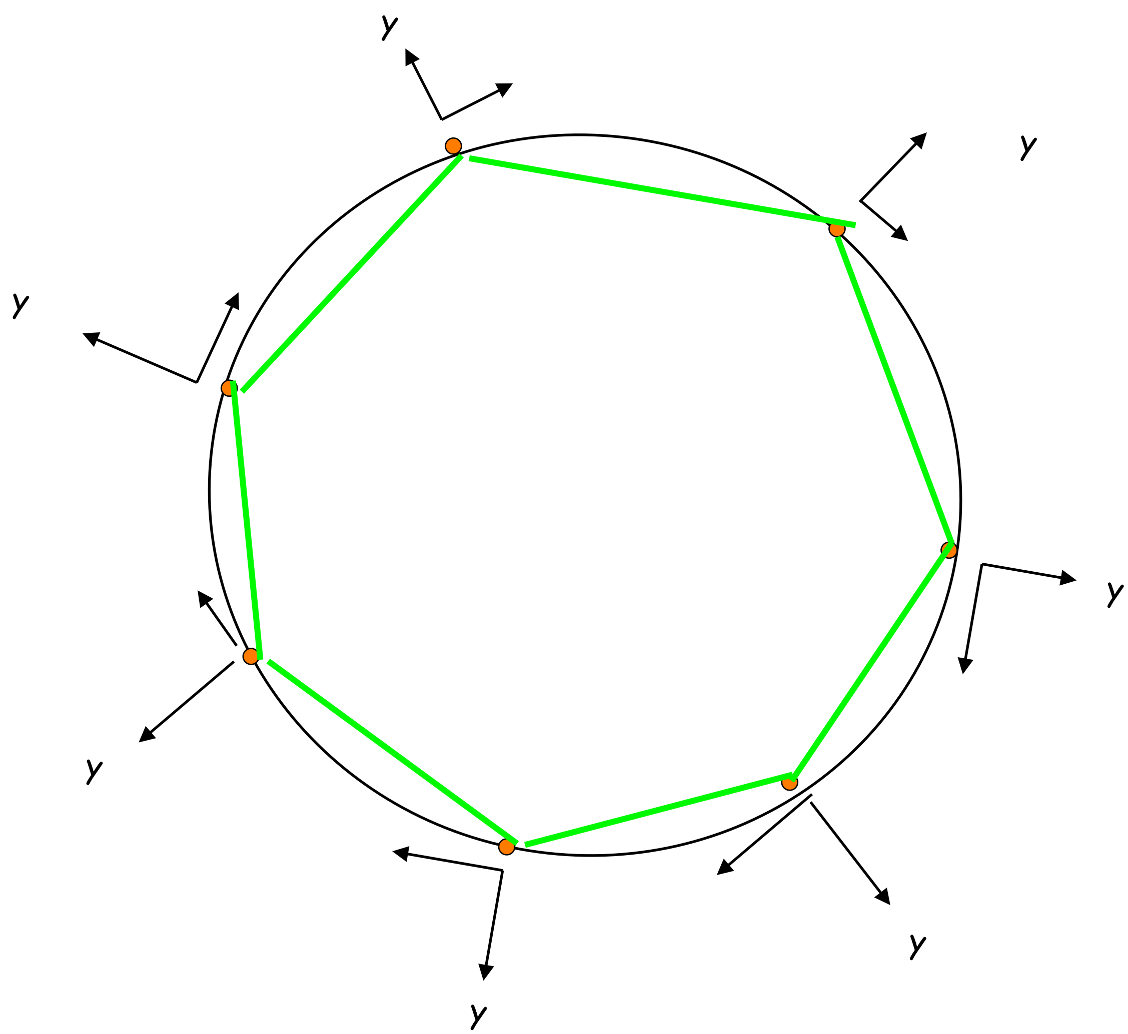
any totally synchronous execution will transform the n-GON into another n-GON



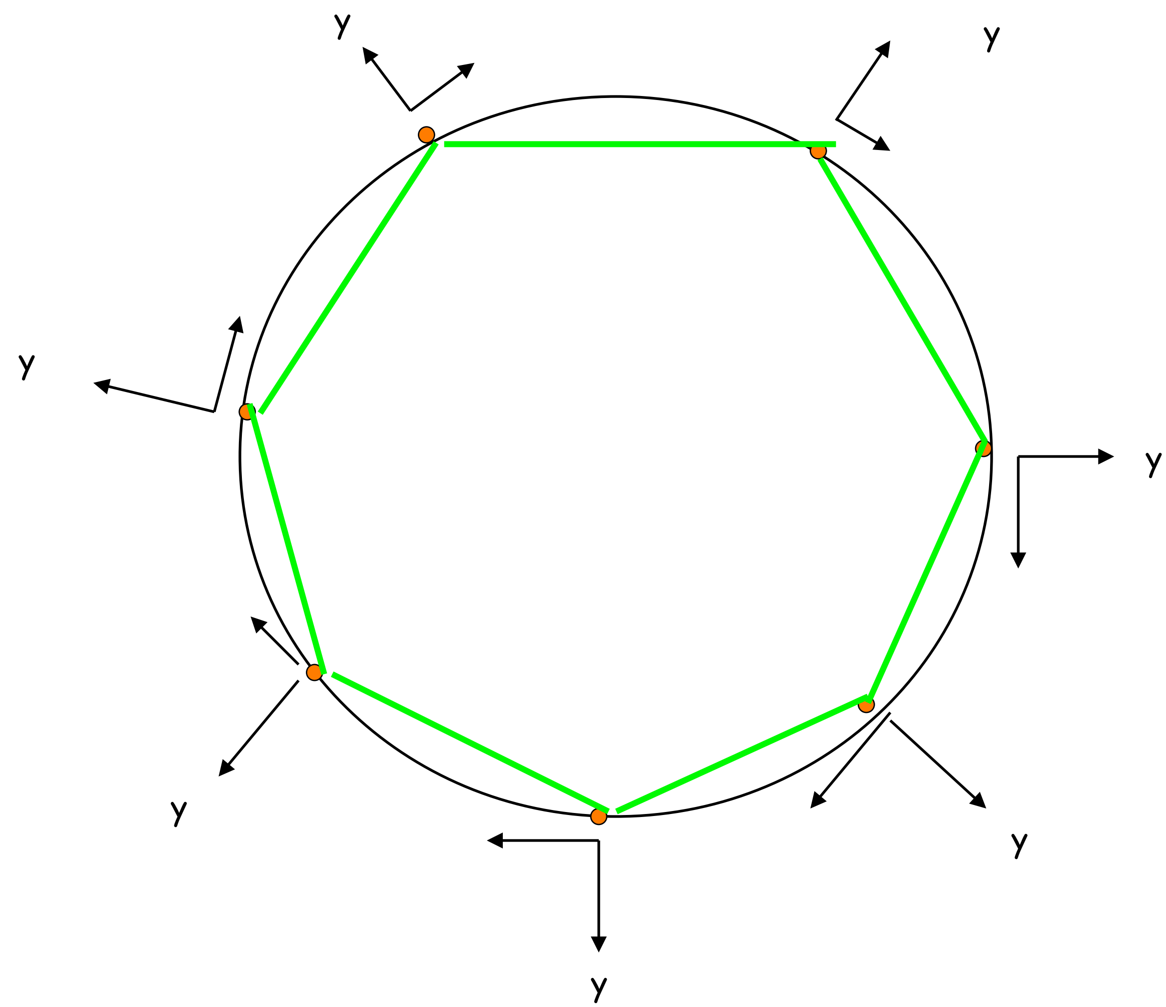
any totally synchronous execution will transform the n-GON into another n-GON



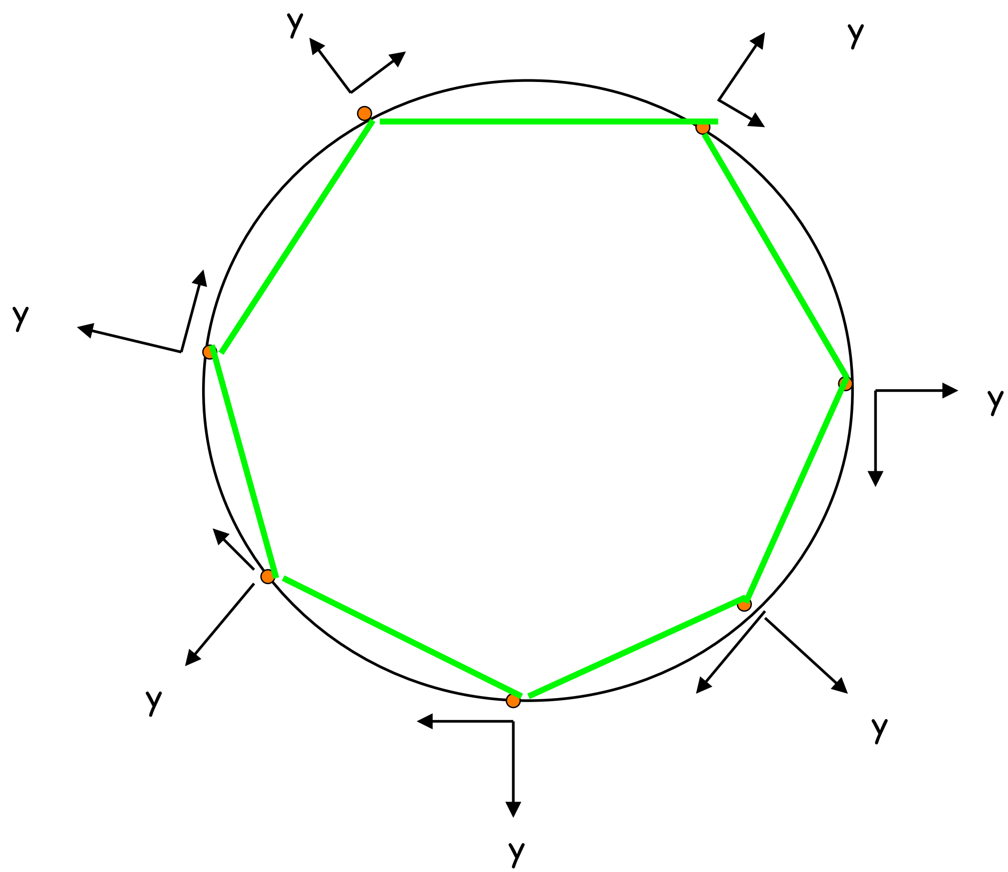
any totally synchronous execution will transform the n-GON into another n-GON



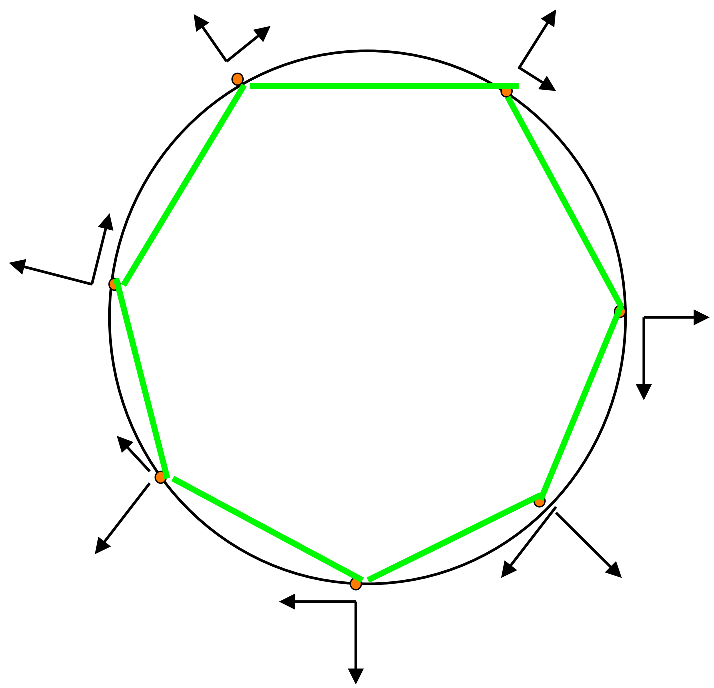
… the only patterns that *can* be formed are n-GON …



… the only patterns that *can* be formed are n-GON …

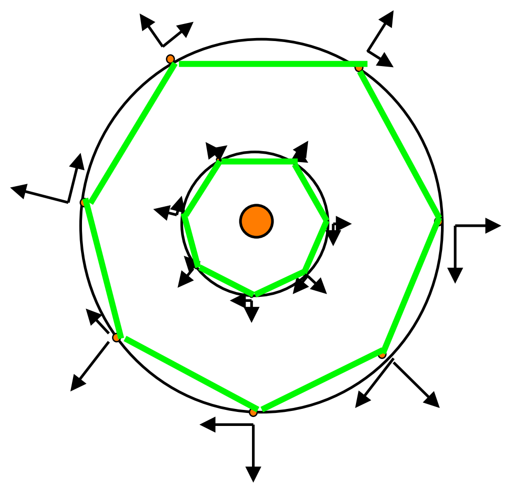


… the only patterns that *can* be formed are n-GON …



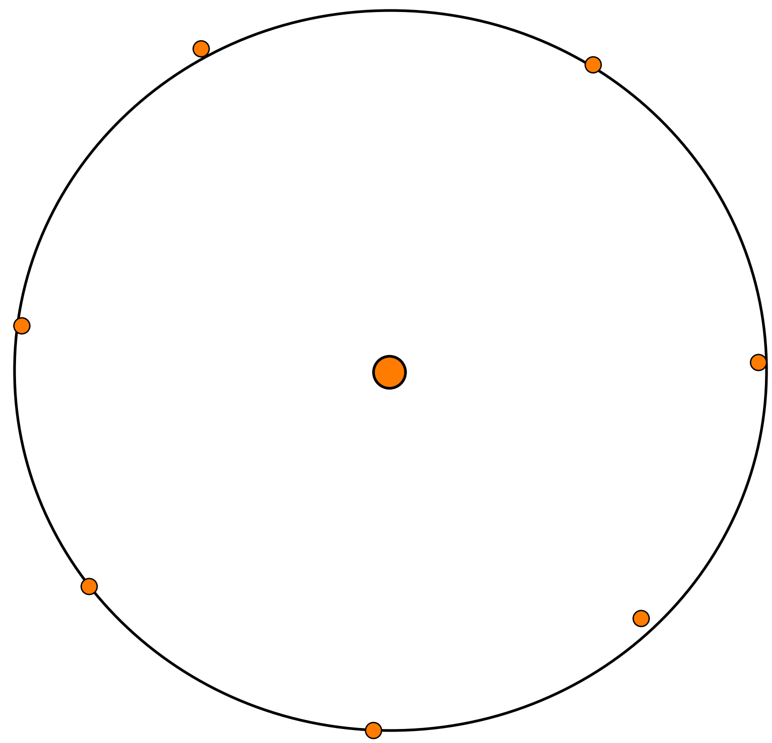
… the only patterns that *can* be formed are n-GON …

… and POINT



… the only patterns that *can* be formed are n-GON …

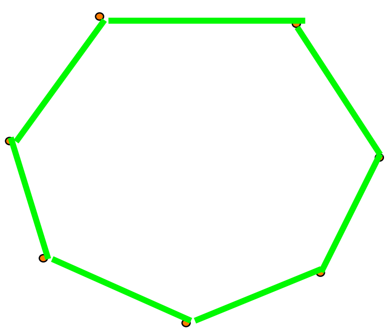
… and POINT



Pattern Formation

No agreement

What patterns can be formed regardless of initial configuration ?



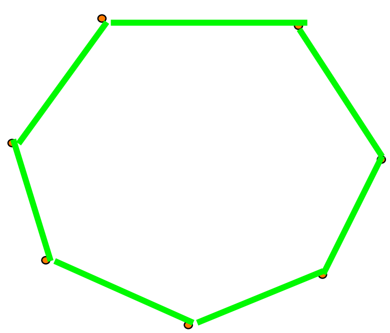
n-GON

POINT

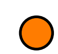
Pattern Formation

No agreement

even if
- **unbounded persistent** memory
- **fully synchronous** system



n-GON

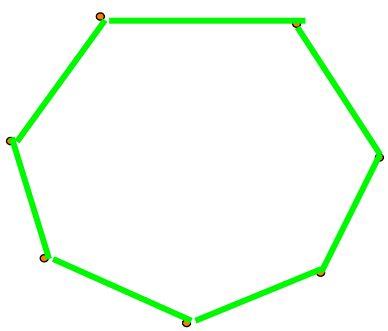


POINT

Pattern Formation

No agreement

can we always form them ?



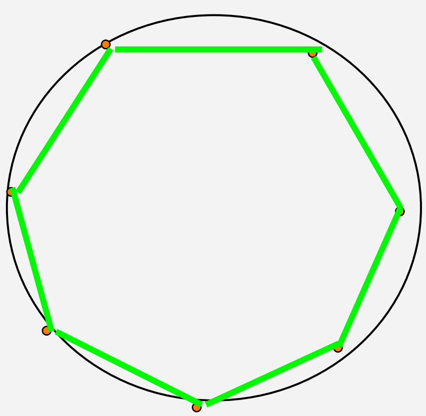
n-GON

POINT

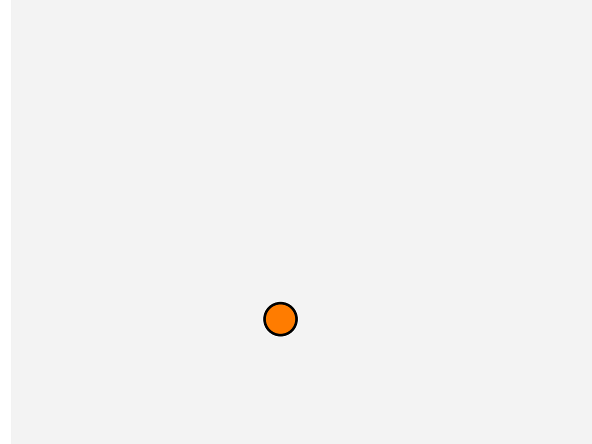
Pattern Formation

No agreement

CIRCLE FORMATION

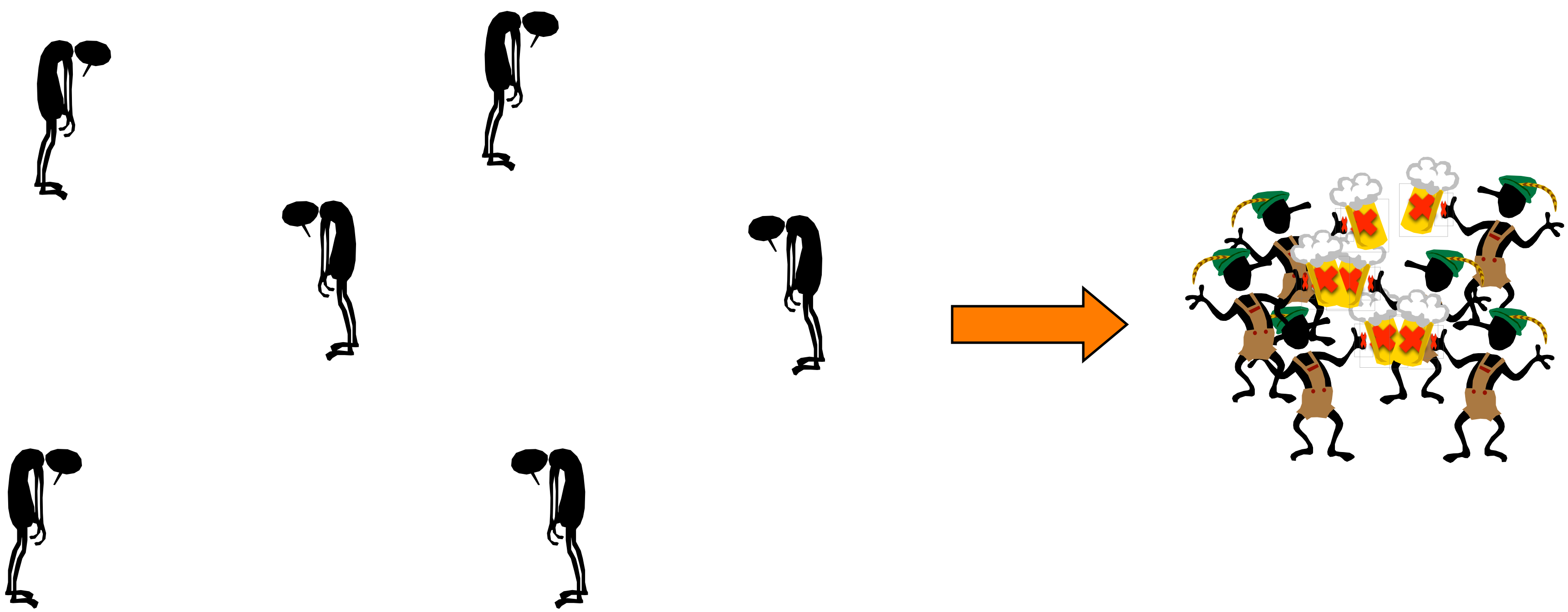


GATHERING



GATHERING

Gathering



Initially the robots are in arbitrary distinct positions.

In finite time, they **gather** in the same place.

Gathering

Convergence:

For any ε there is a time t when the robots are at least ε -close from that time on.

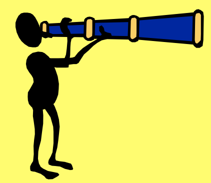
Formation:

In finite time the robots meet exactly in the same point and stay there.

Gathering

- In spite of its apparent simplicity, this problem has many hidden difficulties

Gathering



With Global Visibility

more than 2 robots

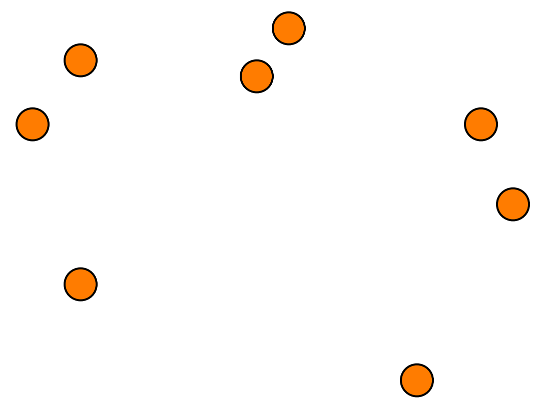
Gathering : simple convergence solution

General approach:

In each cycle, the robots:

1. calculate the center of gravity position of the group
2. move towards that position

$$\vec{c}[t] = \frac{1}{N}\sum_{i=1}^{N}\vec{r}_i[t]$$



Gathering : simple convergence solution

General approach:

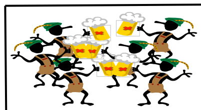
In each cycle, the robots:

1. calculate the center of gravity position of the group
2. move towards that position

The center of gravity changes while the robots move.

but the **convex hull of the robots shrinks**

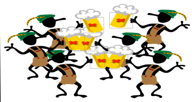
CONVERGENCE of n>2 robots

	Convergence
FSYNCH	Yes
SSYNCH	Yes
ASYNCH	Yes

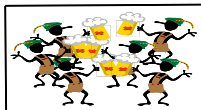
Center of Gravity algorithm

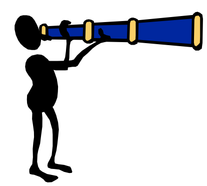
Cohen, Peleg, SIAM J. Comput. 2005

GATHERING of n>2 robots

	Gathering
FSYNCH	
SSYNCH	
ASYNCH	

GATHERING of n>2 robots

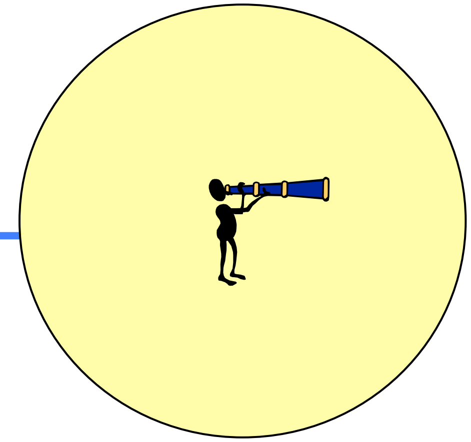
	Gathering
FSYNCH	Yes simple
SSYNCH	Yes simple
ASYNCH	Yes complex



With multiplicity detection

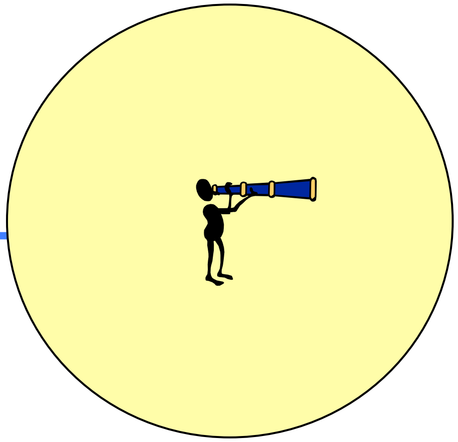
Cielibak, Flocchini, Prencipe, Santoro, SIAM J. Comp 2012

Convergence of n>2 robots



with limited visibility

Convergence of n>2 robots



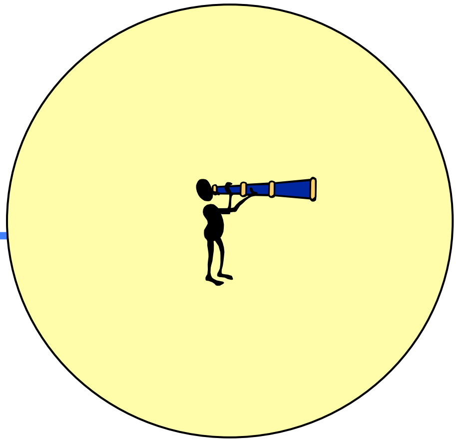
with limited visibility

limited	Convergence
FSYNCH	yes
SSYNCH	YES
ASYNCH	?

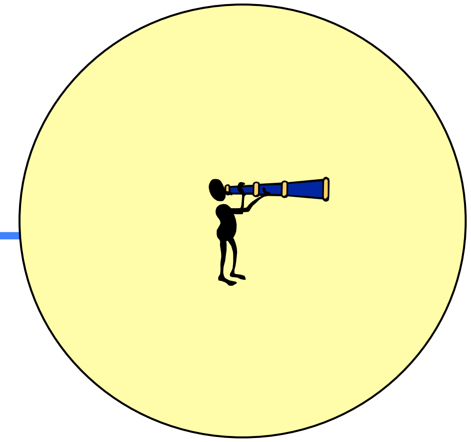
Ando,Oasa,Suzuki,Yamashita, *IEEE Trans. Rob. Aut.* *1999*

GATHERING of n>2 robots

with limited visibility

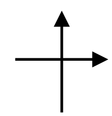


GATHERING of n>2 robots



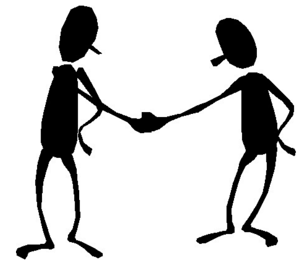
with limited visibility

limited	Gathering
FSYNCH	yes
SSYNCH	yes
ASYNCH	YES



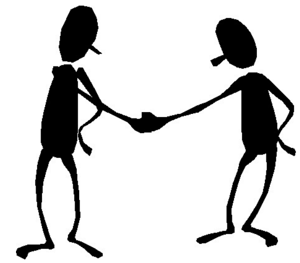
Flocchini, Prencipe, Santoro, Widmayer. *TCS*, 2005

GATHERING of n=2 robots

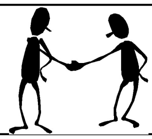


Rendezvous

GATHERING of n=2 robots



Rendezvous

	Convergence	Gathering
FSYNCH	yes	yes
SSYNCH	yes	impossible
ASYNCH	yes	impossible

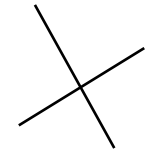
Suzuki, Yamashita, SIAM J. Comp 1999

Computability Implication

FSYNCH > SSYNCH

Suzuki, Yamashita, SIAM J. Comp 1999

Pattern Formation

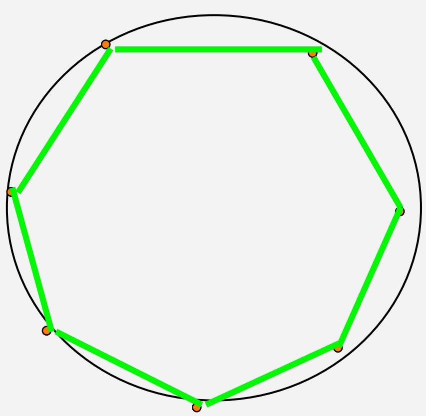


No agreement

Can be done !
ASYNCH Oblivious

Can it be done ?

CIRCLE FORMATION



GATHERING

Cielibak,Flocchini,Prencipe,Santoro,
SIAM J. Comp 2012

UNIFORM CIRCLE FORMATION

Uniform Circle Formation

Partial Solutions

Suzuki, Yamashita, SIAM J. Comp 1999

Défago, Konagaya, WPMC 2002

Souissi, Défago, Katayama, JSF 2004

Chatzigiannakis, Marcou, Nikoletseas, WEA 2004

Katreniak, SIROCCO 2005

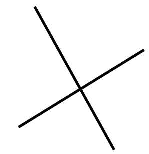
Dieudonne, Petit, *IPL* 2007

Dieudonne,Labbane,Petit, *ACM Trans. Aut. Adap. Sys*. 2008

Dieudonne,Petit, ISAAC 2008

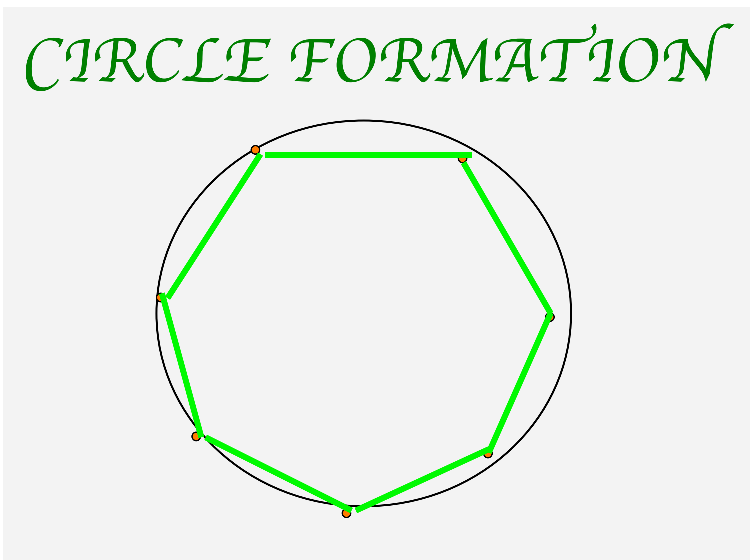
Miyamae.Ichikawa,Hara, *J. Robotics and Mechatr*. 2009

Pattern Formation



No agreement

Can be done !
ASYNCH Oblivious



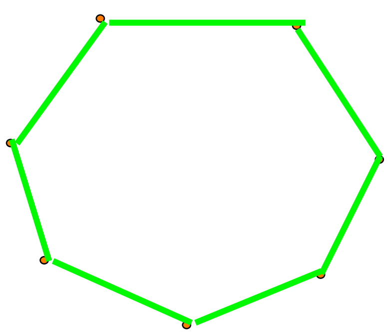
Flocchini,Prencipe,Santoro,Viglietta,
OPODIS 2014

Pattern Formation

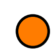
No agreement

can we always form them ?

YES !

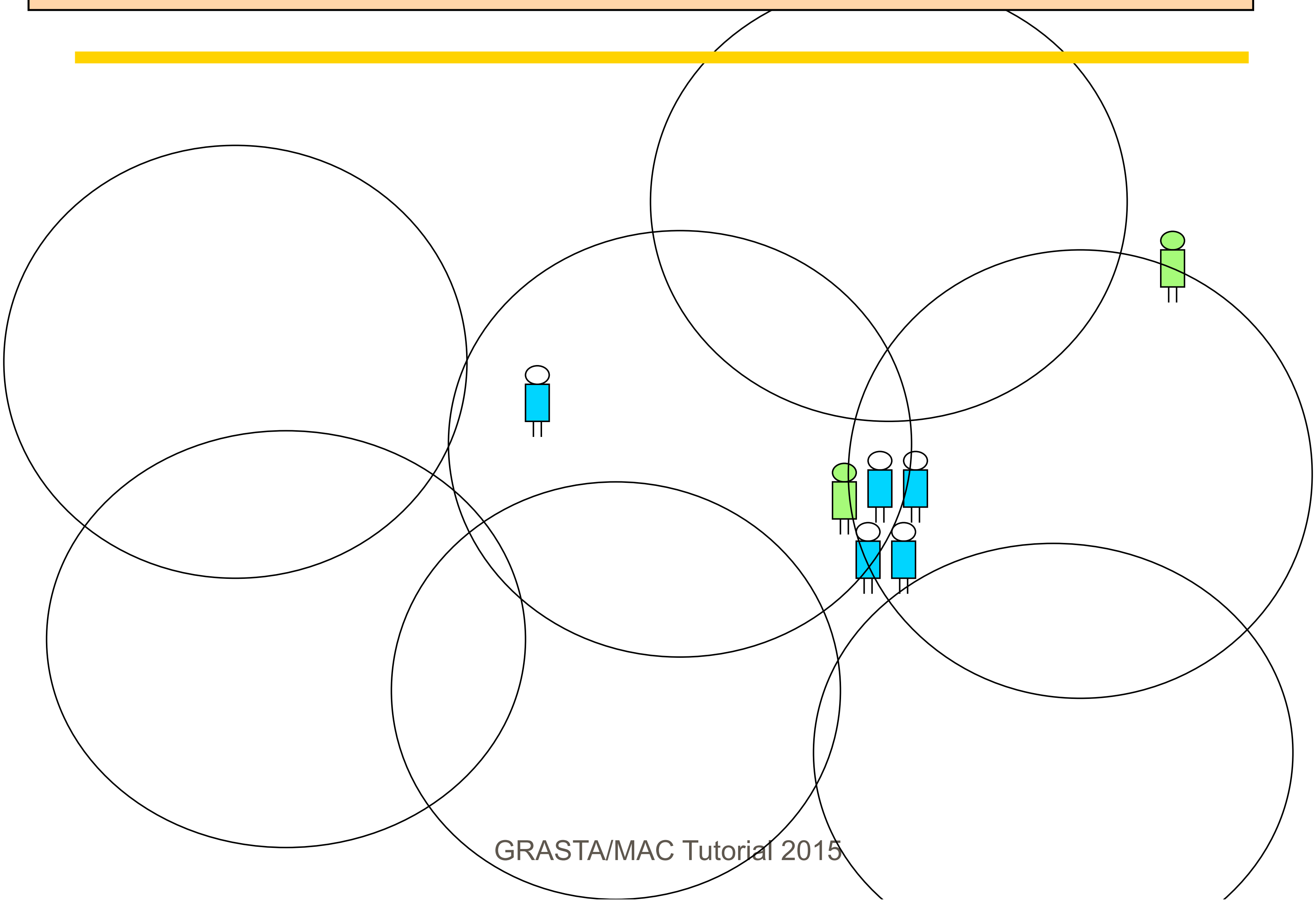


n-GON



POINT

Self-Deployment, Covering, Scattering



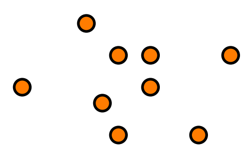
Sensor Networks : Deployment

Deployment : distributing the sensors over the territory so to cover the area according to some optimization criteria (e.g., a uniform coverage)

Mobile Sensor Networks

SELF Deployment : the sensors position themselves over the territory so to cover the area according to some optimization criteria (e.g., a uniform coverage)

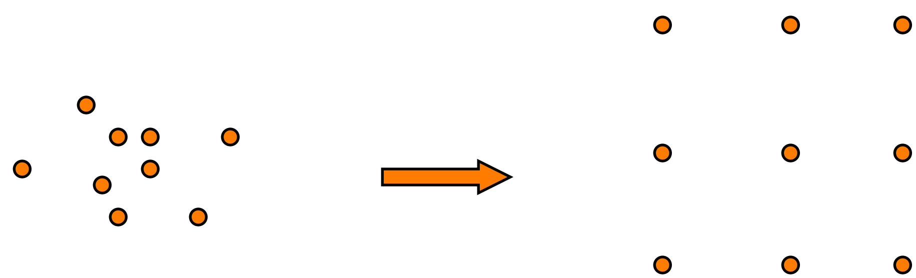
Self-Deployment, Covering, Scattering



Initially the sensors are in arbitrary positions.

In finite time, they **scatter** throughout the territory

Self-Deployment, Covering, Scattering

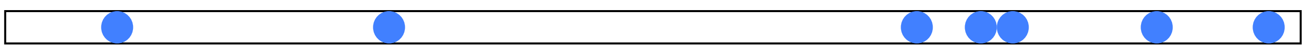


Initially the sensors are in arbitrary positions.

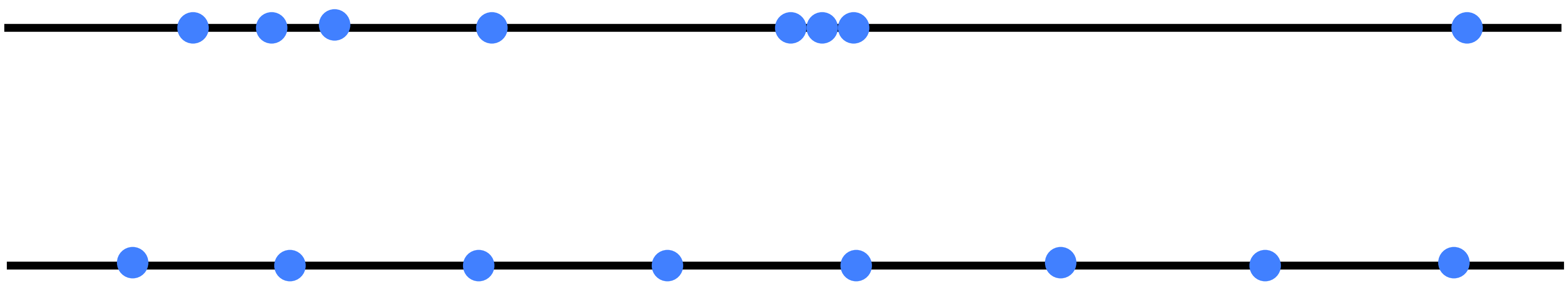
In finite time, they **scatter** throughout the territory

Scattering

On a corridor (or line)

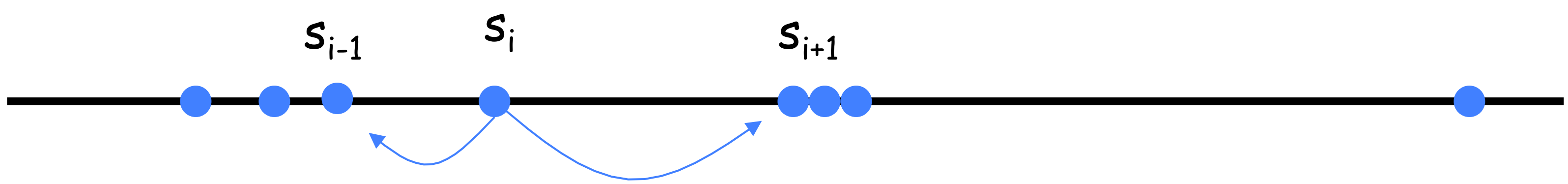


Scattering on the line



- convergence
- semi-synchronous
- difficult proof

Scattering on the line



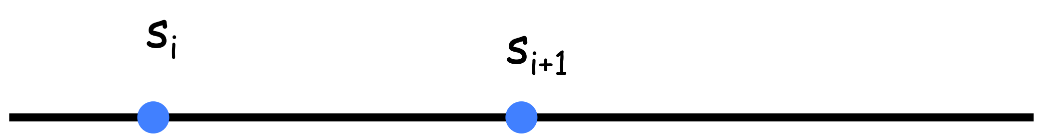
Assumption: a robot sees its neighbouring robots

Scattering on the line

ALGORITHM for robot s_i

go-to-half

If no robot is seen on the left or on the right (I am first or last)
do not move



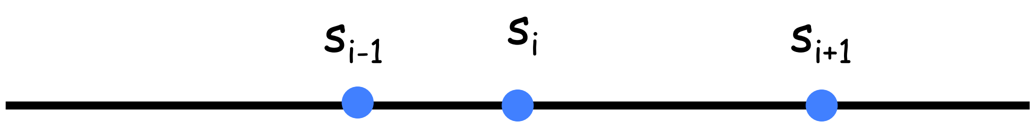
Scattering on the line

ALGORITHM for robot s_i

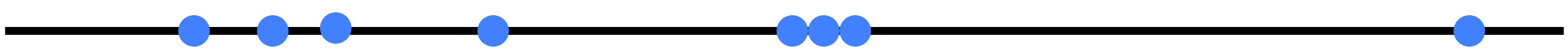
go-to-half

If no robot is seen on the left or on the right (I am first or last)
do not move

Otherwise **move to x = ½ dis(s_{i-1}, s_{i+1})**



Scattering on the line



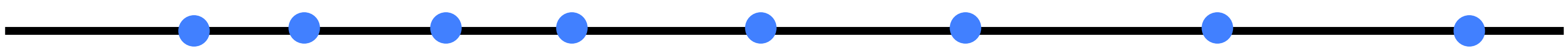
Scattering on the line

Scattering on the line

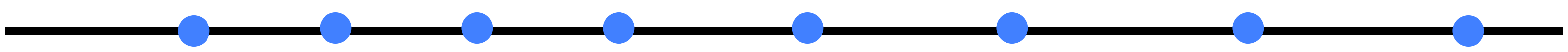
Scattering on the line

Scattering on the line

Scattering on the line



Scattering on the line



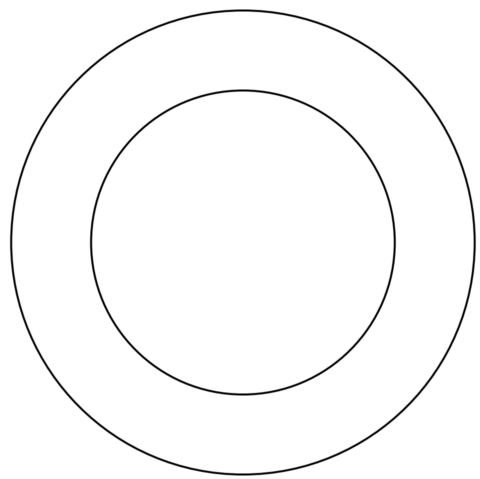
Scattering on the line

- convergence
- semi-synchronous
- difficult proof

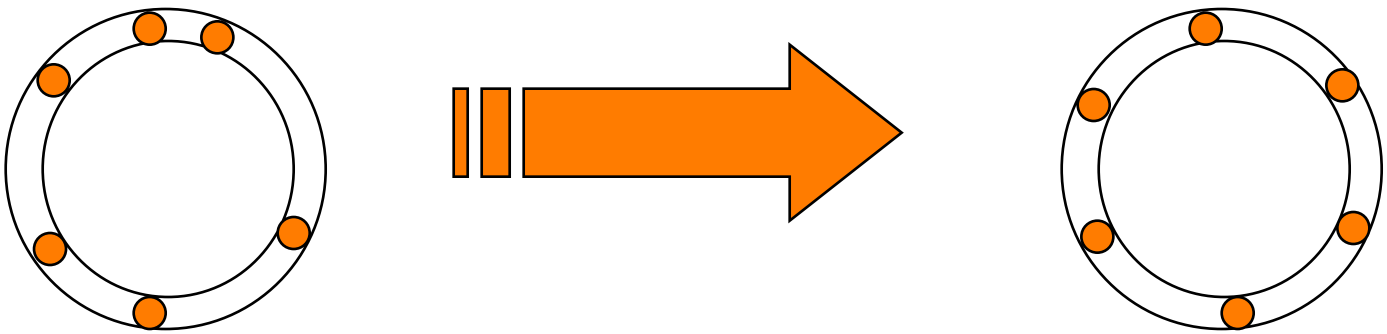
R. Cohen, D. Peleg, *Theoretical Computer Science*, 2008

Scattering

On a circular border (ring)



Scattering on the ring

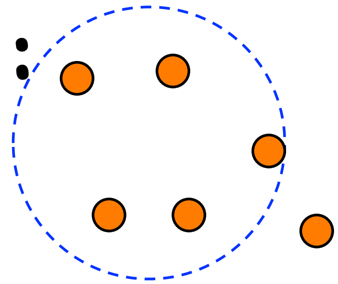


Scattering on a ring

Related to *uniform circle formation*

Scattering on a ring

Differences:

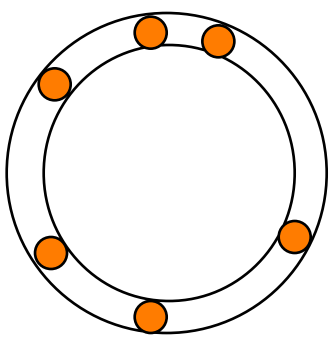


Uniform circle formation

the space is the two dimensional plane and the robots can move and form the circle anywhere in the space

Scattering on ring

the space is the ring and the robots can move only on it



Scattering on a ring

Without common orientation, scattering is **impossible**

even if the sensors have unbounded memory, unlimited visibility, and they are semi-synchronous

With common orientation, scattering is **possible**

even if the sensors are oblivious, have limited visibility, and they are asynchronous

Flocchini, Prencipe, Santoro, *Theoretical Computer Science*, 2008.

Mobile Robots

Advanced settings:

Mobile Robots

Advanced settings:

- compass inaccuracies
- movement inaccuracies
- sensing inaccuracies
- faulty robots

Agmon, Peleg. *SIAM J. Comp.* 2006

Souissi,Défago,Yamashita, OPODIS 2006

Bouzid, Potop-Butucaru,Tixeuil, *TCS* 2010

Souissi,Izumi,Wada, *CCCA* 2011

Yang·Souissi· Défago·Takizawa· J. Syst. and Software 2011

Izumi,Souissi,Katayama,Inuzuka,Defago,Wada,Yamashita. *SIAM J. Comp.* '12

Bouzid.Das,Tixeuil, ICDCS 2013

Mobile Robots

Advanced settings:

- sequence of patterns
- obstructed views
- near gathering
- probabilistic

Yamauchi, Yamashita, DISC 2014.

Das,Flocchini, Santoro,Yamashita. *Distributed Computing* 2015

Pagli,Prencipe,Viglietta, *Distributed Computing 2015*

Mobile Robots

Advanced settings:

- robots with physical dimension
- computing with obstacles
- probabilistic algorithms

Czyzowicz, Gasieniec, Pelc, *TCS* 2009

Bolla,Kovacs,Fazek. ISSEC 2012

Agathangelou, Georgiou, Mavronicolas, PODC 2013

Honorat, Potop-Butucaru,Tixeuil, *TCS* 2014

Gan Chaudhuri,Mukhopadhyay. *J. Discrete Alg.* 2015

Mobile Robots

Advanced settings:

- complexity
  - overall distance travelled (energy?)
  - time

Mobile Robots

Advanced settings:

- 3D space (drones)

Yamauchi,Uehara, Kijima,Yamashita, DISC 2015

Mobile Robots

Advanced settings:

- Luminous robots

Das, Flocchini, Prencipe,Santoro,Yamashita, ICDCS 2012

Flocchini,Santoro,Viglietta,Yamashita, SIROCCO 2013

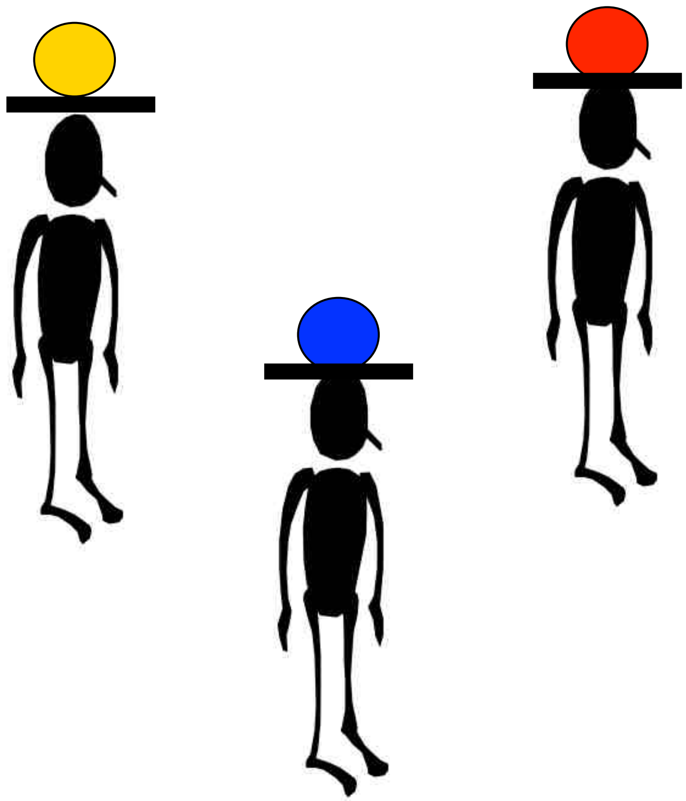
Das,Flocchini,Prencipe,Santoro, FUN 2014

Das, Flocchini, Prencipe,Santoro,Yamashita, *TCS* 2015.

Di Luna,Flocchini,Gan Chaudhuri,Poloni,Santoro,Viglietta, *Inform. Comp.* 2015

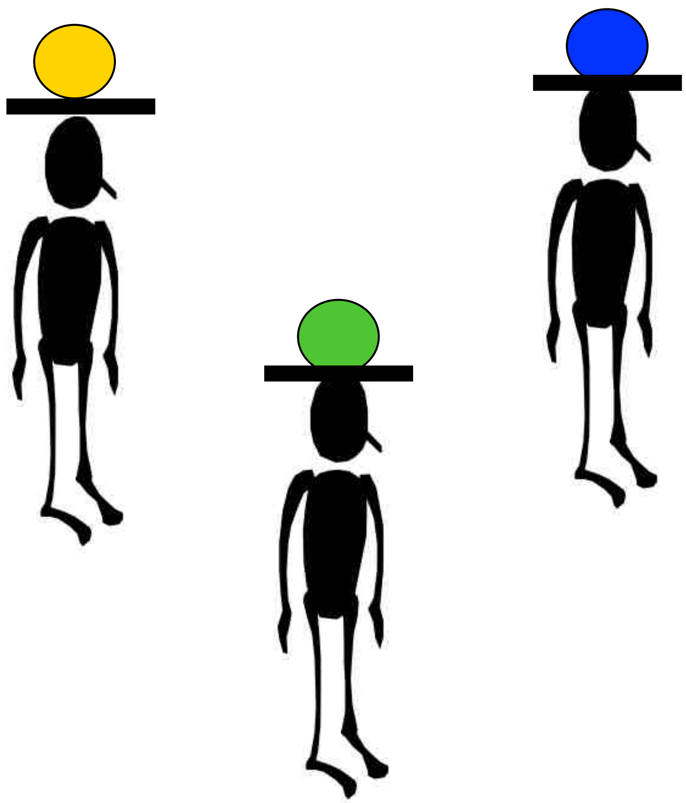
Luminous Robots

The robots are enhanced
with **VISIBLE LIGHTS**
that can change **color.**

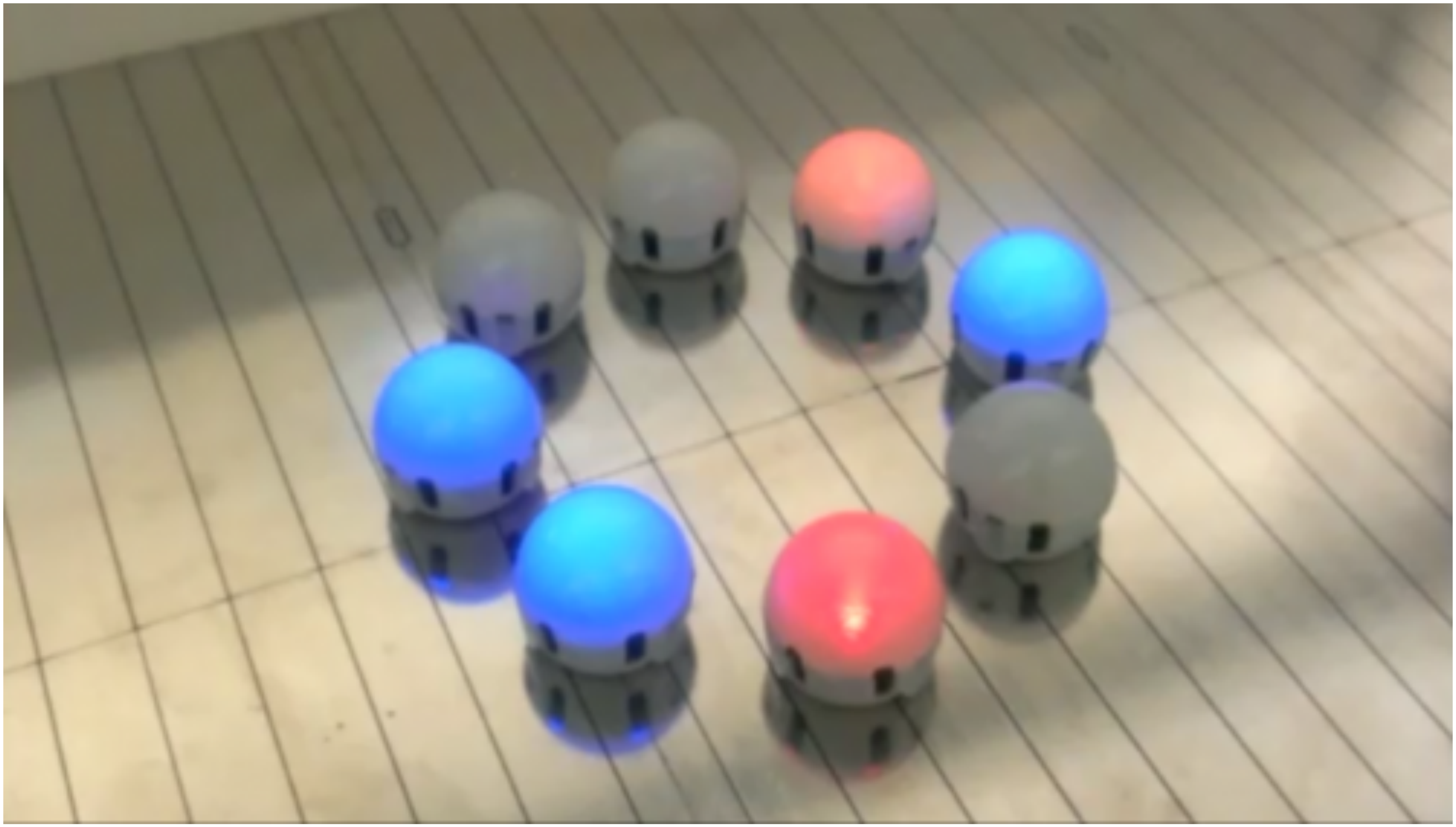


Luminous Robots

The robots are enhanced
with **VISIBLE LIGHTS**
that can change **color.**



Luminous Robots

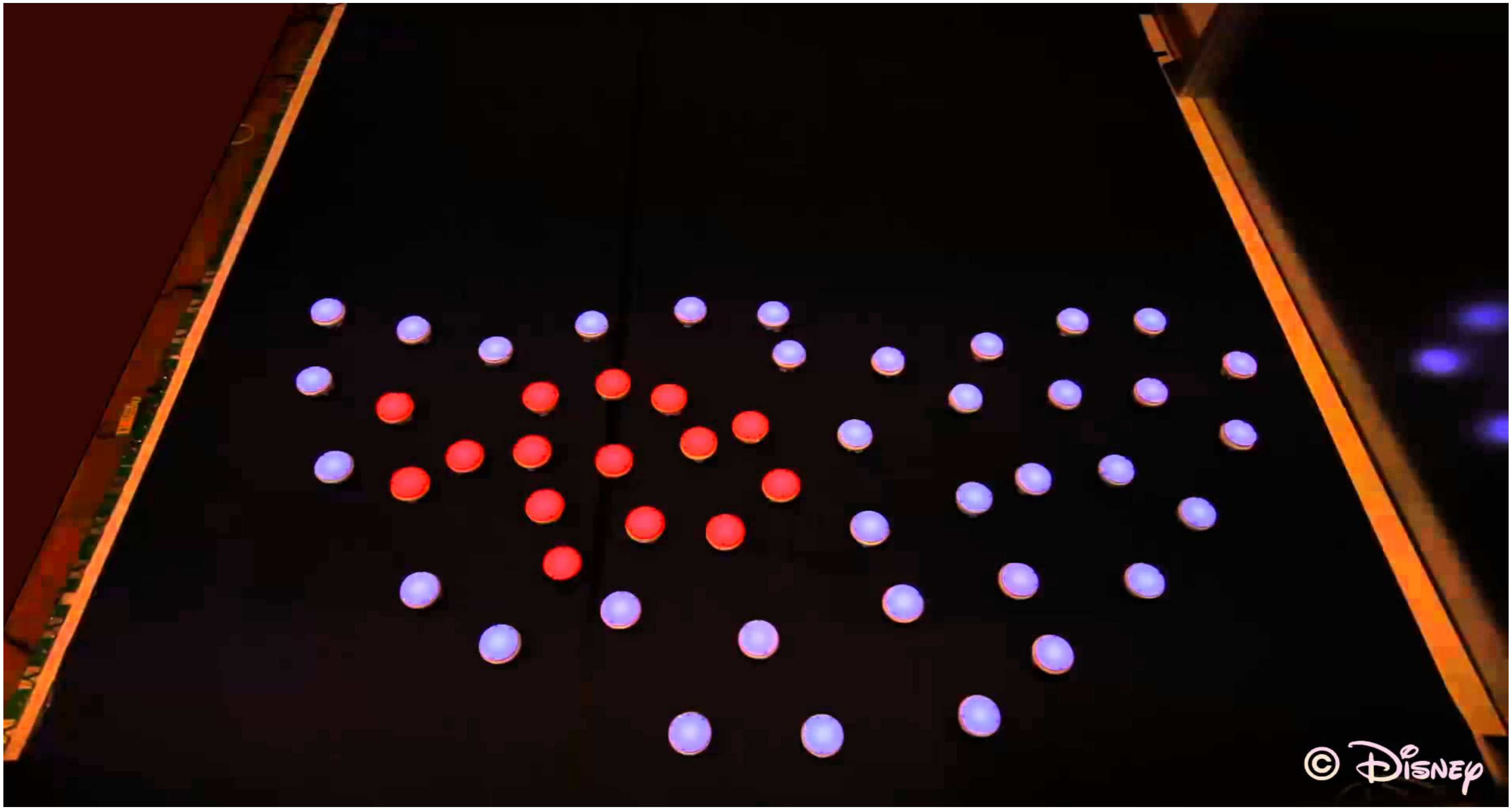


Droplets [U.C. Boulder]

Luminous Robots

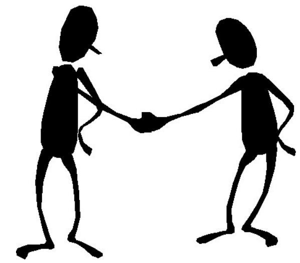
Anarchitecture [Dorigo]

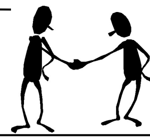
Luminous Robots



Pixelbots [Disney & ETH]

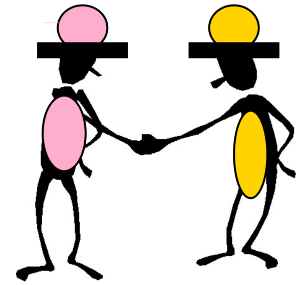
Gathering of 2 robots without lights

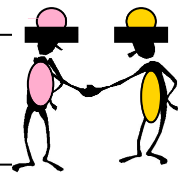


	
FSYNCH	yes
SSYNCH	impossible
ASYNCH	impossible

Suzuki, Yamashita, SIAM J. Comp 1999

Gathering of 2 robots with lights

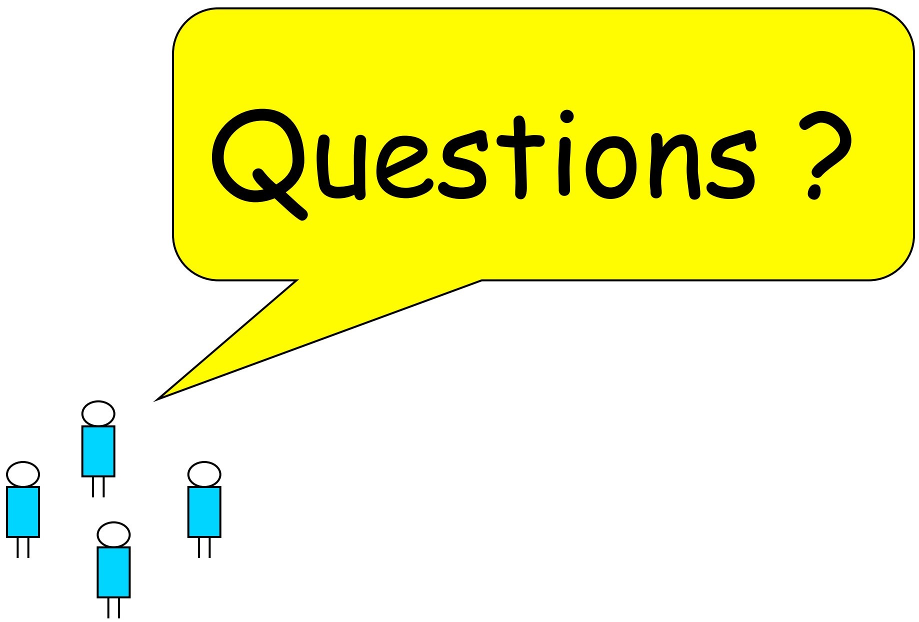


	
FSYNCH	yes
SSYNCH	yes
ASYNCH	yes

Viglietta, ALGOSENSORS 2013
(2 colors, optimal)

(4 colors)

Das, Flocchini, Prencipe,Santoro,Yamashita, *TCS* 2015


Questions ?